Sentiment Analysis for the Low-Resourced Latinised Arabic "Arabizi"

Thesis

Sentiment Analysis for the Low-Resourced Latinised Arabic "Arabizi"

Taha Tobaili

A Thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy (PhD)

Knowledge Media Institute
The Open University
UK

August 2020

Supervisors:
Dr. Miriam Fernandez and Prof. Harith Alani

Dedication

Whilst finalising this thesis a humungous blast devastated my hometown Beirut that took the world by surprise. It was estimated to be the second largest explosion in history. Lebanon was once named Switzerland of the east for its unique natural beauty in the region is now severely damaged. The blast took over 200 lives, injured 5,000 citizens, and destroyed 300,000 homes so far. I hereby partially dedicate this thesis to every mother and father that lost their child, every sister and brother that lost their sibling, and every person that lost their parent, friend, relative or lover.

Abstract

The expansion of digital communication mediums from private mobile messaging into the public through social media presented an opportunity for the data science research and industry to mine the generated big data for artificial information extraction. A popular information extraction task is sentiment analysis, which aims at extracting polarity opinions, *positive*, *negative*, or *neutral*, from the written natural language. This science helped organisations better understand the public's opinion towards events, news, public figures, and products.

However, sentiment analysis has advanced for the English language ahead of Arabic. While sentiment analysis for Arabic is developing in the literature of Natural Language Processing (NLP), a popular variety of Arabic, Arabizi, has been overlooked for sentiment analysis advancements.

Arabizi is an informal transcription of the spoken dialectal Arabic in Latin script used for social texting. It is known to be common among the Arab youth, yet it is overlooked in efforts on Arabic sentiment analysis for its linguistic complexities.

As to Arabic, Arabizi is rich in inflectional morphology, but also codeswitched with English or French, and distinctively transcribed without adhering to a standard orthography. The rich morphology, inconsistent orthography, and codeswitching challenges are compounded together to have a multiplied effect on the lexical sparsity of the language, where each Arabizi word becomes eligible to be spelled in many ways, that, in addition to the mixing of other languages within the same textual context. The resulting high degree of lexical sparsity defies the very basics of sentiment analysis, classification of positive and negative words. Arabizi is even faced with a severe shortage of data resources that are required to set out any sentiment analysis approach.

In this thesis, we tackle this gap by conducting research on sentiment analysis for Arabizi. We addressed the sparsity challenge by harvesting Arabizi data from multi-lingual social media text using deep learning to build Arabizi resources for sentiment analysis. We

developed six new morphologically and orthographically rich Arabizi sentiment lexicons and set the baseline for Arabizi sentiment analysis on social media.

Content

Acknowledgement

I write this as a chapter of my life comes to an end, a four-year journey of knowledge that enlighted me with different disciplines, took me to many places, and connected me with bright people that had a great influence on the person I have become.

I start by sharing my sincere appreciation to the ones whom this work would not have been made possible without their superlative supervision and friendly character Dr Miriam Fernandez and Prof Harith Alani.

Thank you Miriam and Harith for trusting me to study a topic of my interest, that was once an abstract and today it is a success story. Thank you Miriam for your intensive academic care from guidance to endless support. Thank you for transferring your sharp skills in planning out research and positioning it finely on paper.

I also thank our lab, the Knowledge Media Institue for supporting me on academic trips to gain further academic experience and present my work at various conferences and academic institutions. Lest I forget, the Open University UK for funding my PhD.

Throughout this journey, I was fortunate to live and study in Germany for one year, where I worked as an NLP intern at IBM Deutschland and visited the NLP focus group at the University of Mannheim. For that, I also thank Prof. Goran Glavas who placed a great impact on my research.

I am also grateful for the generous time and effort contribution of Chris Sanders for developing and maintaining the project's system and Rana Islambuli, Omar Farhat, and Omar Osman for developing the datasets integral to this research.

Finally, I am forever grateful to my beloved mom, dad, and grandmother for their significant patience, prayers, and faith. I extend this gratitude with my siblings, friends, and relatives for their share of moral support.

Preface

I started this research with the intent of studying sentiment analysis for Arabizi only to find myself drowned in the linguistic complexities of Arabizi. Unlike traditional theses, this research became more applied than theoretical.

As I study the natural language found on social media, I decided to express my natural narrative in the writing of this thesis without relying on a spell check in case you came across some typos.

I start every chapter with pieces of Arabic poetry and sayings that shows the charm of derivational morphology and semantics. Translating them to English loses their phonemic, morphological, and morphophonemic beauty, hence I left them intact.

The following poetry expresses sincere love for the Arabic language, emphasising on how (unlike many other languages) Classical Arabic survived through the ages that it is read and understood to our day.

لا تلمني في هواها أنا لا أهوى سواها

Don't blame me in loving her, for I love nothing but her

لست وحدي أفتديها كلنا اليوم فداها

I am not the only sacrifice for her, rather today we all are

نزلت في كل نفس وتمشّت في دماها

She resided in every self, and crawled through their blood

فبها الأم تغنّت وبها الوالد فاها

Through her the mother sang
And through her the father spoke

وبها الفن تجلى وبها العلم تباهى

And through her the art was made clear
And through her the art showed off

كلما مرّ زمان زادها مدح وجاها

Every time an age passes, it increases her in glory and value

لغة الأجداد هذي رفع الله لواها

This is the language of the ancestors, God has raised its flag

فأعيدوا يا بنيها نهضة تحيي رجاها

So repeat o its children, a spring the revives its glory

لم يمت شعب تفاني في هواها واصطفاها

A nation that puts effort in loving and purifying her never dies

Halim Dammous

لأُمّى

To My Mother

Introduction

صادق صديقا صادقا فى صدقه فصدق الصداقة فى صديق صادق

1.1 Motivation

Arabic is the language to over 420 million people making it the fifth most spoken language in the world[1]. With a remarkable growth of social networking in Arab speaking countries, Facebook stated that the network has 164 million active monthly users in the region (Radcliffe & Bruni, 2019). Amidst all the major events taking place in the Middle East and North Africa, from wars and invasions to political conflicts and uprising of the nations, people utilised the social media to express their sentiments publicly towards the events and turmoil surrounding them. The availability of such daily-generated plethora of digital data that represents the peoples' voices presented an opportunity for the data science community to exploit for Artificial Intelligence (AI) mining and analysis. This promotes individuals, governments, and organisations a fast and effective way to monitor the public's opinion, understand social behaviour, and predict the people's reaction towards imminent events. It also allows companies to understand the public to cater products and services to their desires.

Sentiment analysis is the study that uses AI to identify positive, negative, and neutral opinions from natural text. It is at the forefront for the English language but still developing for Arabic. Arabic is the official language for 24 countries; though widely spoken, it has several varieties. Modern Standard Arabic (MSA) is the formal form of the language, written and spoken, structured extensively by linguists since centuries, and standardised across the Arab region. Dialectal Arabic (DA) is the informal spoken form of Arabic, esoteric to each Arab region, differs in word choice, morphology, pronunciation, and speech tempo hence lacking a standard orthography. During the rise of digital texting in the Arab world, a new linguistic phenomenon was born, the transcription of the spoken dialectal Arabic in Latin script, known as Arabizi.

[1] http://istizada.com/complete-list-of-arabic-speaking-countries-2014/

Studies reported that over 60% of digital communication is Arabizi in E-mail and mobile messaging within some Arab communities (Aboelezz, 2009), (Bies, et al., 2014). It is a common way of communication among the youth (Bhandari, 2018), (Keong, et al., 2015), (Allehaiby, 2013), (Muhammed, et al., 2011) and proven to be a key communication medium in relevant events in the Arab world such as the Arab spring (Basis-Technology, 2012) yet it is overlooked in the literature of Arabic sentiment analysis (Duwairi & Qarqaz, 2014), (Al-Kabi, et al., 2013), (Al-Kabi, et al., 2014). It is inconsistent in orthography and suffers from a scarcity of Natural Language Processing (NLP) resources. This motivated us to research sentiment analysis for Arabizi on social media.

We initiate this thesis with a statistical analysis on the usage of Arabizi on social media across two Arab regions. We scrutinise how the users naturally Latinise Arabic without a consensus on a writing system to reveal the underlying complexities that pose challenges to sentiment analysis. We then propose to utilise a deep learning approach to address these challenges and develop new Arabizi resources for sentiment analysis. Since Arabizi is common among the youth and Lebanon ranks first among the most active social networking countries for younger users with 90% of its social media users aging between 18 and 36 (Radcliffe & Bruni, 2019), and is known for its bilingualism (Shaaban, 1997), we chose the Lebanese dialect Arabizi as the use case for this research.

1.2 Research Questions

Given the mentioned observations, the main research question we investigate in this thesis is:

How to analyse sentiment from Arabizi text?

With the heavy linguistic complexities present in Arabizi, the main goal that drives this thesis is to explore which approaches could be used to analyse sentiment from Arabizi text.

Before we initiate the briefest investigation in analysing sentiment from Arabizi text, we reason why we haven't focused our efforts on de-Latinising Arabizi, transliterating it to

Arabic, prior to sentiment analysis, since it is a human transliteration of Arabic into Latin script by nature.

Arabizi is a reflection of the spoken DA in Latin script, hence the way users transcribe Arabizi is inconsistent and different among different regions. Users follow some orthographic standards that are normalised to a small extent within their regions such that Egyptian Arabizi differs from Levantine Arabizi not only by dialect but also by the choice of letters and the style of mapping the Arabic phonemes with the Latin script. Even slightly normalised orthography within regions is inconsistent, that one might spell the same word differently at different times. Also, the Latin script letters in Arabizi correspond to a wider range of Arabic letters. This generates word ambiguity for Arabizi transliteration, discussed in detail in Chapters 2 and 3. However, even if the text is to be transliterated to Arabic, at best it will result in dialectal Arabic (Callison-Burch, et al., 2011), another low-resourced language for sentiment analysis, because Arabizi is a transcription of the spoken DA not the formal MSA.

Sentiment analysis in its very basic form aims to detect the general polarity of a text fragment: positive, negative, or neutral. Common ways to achieve this include using unsupervised and supervised methods (Zhang, et al., 2018) also known as lexicon-based and machine learning (ML) approaches (Liu, 2012).

Lexicon-based approaches are based on the use of sentiment lexicons. A sentiment lexicon is a list of words associated with sentiment classes such as (positive, negative, neutral) or sentiment scores such as (*love 0.88, hate -0.76*). A lexicon-based approach matches the words of the lexicon with those of the text, assigns the associated classes or scores to the matched words, and finally computes an overall score or deduce a sentiment class of the given text. Therefore, the quality of the lexicon or the correctness of the associated classes directly impacts the accuracy of the sentiment classification (Chapter 3).

ML approaches are trained from pre-labelled text (*positive, negative* for example) to learn the most discriminative features in each sentiment class. Based on this learning process, ML approaches become able to determine the sentiment of new unlabelled text. In other words, ML classifiers learn by example (Chapter 3).

The advantage of the lexicon-based approach is the ability to trace errors as a result of wrongly classified words to the lexicon and hand-fix them. This flexibility of modifying the lexicon makes the lexicon-based approach maintainable and easily improved for continuous development. However, it falls short in classifying contextual words, words that could give different meanings in different contexts, and classifying positive or negative sentences that lack sentiment words. ML approaches are artificially smarter on this front; it is possible that they could learn such patterns but at the high cost of labelling sufficient data to train them. Though the features that ML approaches learn from can be engineered, the classification mechanism is hidden, hence tracing wrongly classified words is not as straightforward as the lexicon-based approach. Also, a general lexicon may perform similar classification on different data domains, as for ML approaches, they might perform well on a dataset of a specific domain but not as well on dataset from a different domain (Chapter 3).

The sentiment expressed in text is inferred from the resulting meaning of the words that comprise the text. As such, both the lexicon-based and the ML approaches look for the vocabulary of the language for classification. The inconsistent orthography of the natural transcription of Arabizi makes its vocabulary possibly way higher in sparsity than Latin script languages with standard orthographies, because a single Arabizi word could be written in various ways. This results in a profound challenge for both sentiment analysis approaches, to encompass all variants of the sentiment vocabulary.

In the light of the current deep learning era, we propose to designate the lexicon-based approach for the research conducted in this thesis with the added advantage of exploring a deep learning technique to build a new orthographically-rich sentiment lexicon to address the sparsity challenge. Therefore, the major contributions here are, a new sentiment resource for Arabizi and an approach that utilises deep learning for building lexical resources, which might be useful for resourcing other low-resourced or highly sparsed languages.

As such, given the mentioned challenges topped by the lack of resources for Arabizi, we aspire to contribute to the Arabic NLP by building new resources for the sentiment analysis of Arabizi.

We present our research questions below:

RQ1: How frequently is Arabizi used on social media and what makes it a challenge for sentiment analysis?

Most sentiment analysis papers that target MSA or DA text simply discard Arabizi although it could represent a considerable portion of the society on social media (Chapter 3). To the best of our knowledge, there are only a handful of papers that focus on Arabizi in NLP. Majority of these works proposed transliterating it to Arabic. We review these papers and point out the pitfalls of transliterating Arabizi in Chapter 3 after explaining the challenges posed by Arabizi in Chapter 2. Earlier works in linguistics investigated the frequency of Arabizi in mobile messaging (Chapter 3) but not on social media.

As opposed to the usage of Arabizi in private mobile messaging, we aim to understand how frequently Arabizi is used by the public on social media, particularly Twitter, with respect to other languages. The frequency of Arabizi differs across Arab regions and probably across digital platforms as well. Thus exploring how often Arabizi occurs within the data streams of Arabic and other languages is important to understand the value of analysing sentiment from the Arabizi segment of these data streams.

Arabizi is known to be a way of communication among the youth, hence the percentage of Arabizi data could represent the voices of communities of interest within the overall Arab social media population.

As such, before ingressing into the pipeline of sentiment analysis, we present a pilot study to assess the volume of Arabizi data generated in Twitter streams across two Arabic speaking countries (Chapter 2).

We follow this study by an investigation of the challenges that this new type of written language poses for sentiment analyses. Arabizi inherits the complexities of the Arabic language, but also introduces additional challenges that are derived from the lack of a standard orthography. In Chapter 2, we detail the Latinisation of Arabic in social text and the challenges that this transcription present for sentiment analysis.

RQ2: How could an Arabizi sentiment lexicon be developed and used for sentiment analysis?

There are two common types of lexicons used in sentiment analysis: One that contains two lists of positive and negative words exclusively like (Hu & Liu, 2004) and others that encompass words exhaustively including neutral words but with assigned polarity scores to each word like Sentiwordnet (Esuli & Sebastiani, 2007).

Such lexicons are usually produced using one or more of the following approaches:

1. Translating other sentiment lexicons i.e. transferring the sentiment words of one language onto another (Chapter 3).

2. Measuring the strength of association between a positive or a negative word with a given set of words to determine their polarity. The pointwise mutual information (PMI) is a known measurement of association among words in NLP (Church & Hanks, 1990). It measures the probability of two words to co-occur in a given corpus. Such that if a known positive word co-occurs frequently with another word, it assigns a positive score to that word. This method has been used to generate several sentiment lexicons (Al-Twairesh, et al., 2016), (Kiritchenko, et al., 2014), (Turney, 2002).

3. Selecting important words from a sentiment labelled dataset. One way of doing this is ranking all words in a sentiment labelled corpus using term frequency-inverse document frequency (TF-IDF), a metric that shows the importance of words to a document, and selecting the sentiment ones among the highest ranked words with the intuition that sentiment words should occur among the most important words to a polar text (Chapter 3).

4. Annotating a large list of random words in a language manually without any previous knowledge about these words. The more human annotators agree on the sentiment of the word the more likely this word is to be accurately annotated. A popular approach is to have three annotators label a list of words, then the words that two annotators agree on their sentiment would be selected for the sentiment lexicon (Chapter 3).

We propose to build an Arabizi sentiment lexicon from ground zero by combining the approaches of 1 and 4 (Translation and Annotation) to produce a sentiment lexicon composed

of two lists of positive and negative words exclusively. The incentive for choosing the translation and annotation approaches is their independency of an expensive sentiment-labelled data and their strength in identifying which words are positive or negative. We also created a sentiment-annotated Arabizi dataset from social media data to evaluate the effectiveness of the proposed lexicon in identifying the sentiment of social media posts.

RQ3: Could word-embeddings enhance the performance of Arabizi sentiment analysis?

A fundamental challenge for word classification rises from the natural Latinisation of Arabic script and its inconsistent orthography. If a sentiment word is written in one orthographic form in the lexicon, how can the sentiment analysis approach match this word with its different orthographic forms appearing in the text?

Since there are no standard rules to map Arabic phonemes with Latin script, each word can be written in numerous ways. In Chapter 2, we explain how trying to capture these variants (written forms of words) in a limited set of patterns to generate them computationally is simply so difficult. A reverse approach to reduce these variants into a single word is as difficult as well.

As such, instead of trying to computationally match words written differently than the ones in the lexicon, we propose to explore word embeddings, a deep learning approach, to retrieve the naturally written forms of the sentiment words from a large compilation of social media texts.

Our plan to create the sentiment lexicon thus comes down to two phases:

1. Generation: Generating a new set of Arabizi sentiment words (RQ2).
2. Expansion: Retrieving the natural variants of the generated sentiment words (RQ3).

Word embeddings is a neural network architecture that converts a large compilation of unsupervised text, naturally occurring (unlabelled), called a corpus, into a space of vectors, where each vocabulary unit of that corpus gets projected as a vector of real numbers.

The position of the word vectors in the embedded vector space depends on how the word embeddings model is tuned. If its tuned for word similarity, it projects words of similar

meaning near each other in the space. This projection of words into vectors positioned by word similarity leveraged language models in the science of NLP, because it opened a space for arithmetic calculations between words from natural text.

If this approach proved to be capable of retrieving the orthographic forms of the proposed Arabizi sentiment lexicon, then hypothetically the lexicon-based approach should cover a wider range of sentiment words, thus the performance of sentiment analysis would improve as a result.

The requirement for word-embeddings is as mentioned a corpus of Arabizi text. Therefore, we develop an Arabizi corpus and expand the proposed sentiment lexicon using the word embeddings deep learning approach. We finally evaluate the effectiveness of the expansion in identifying the sentiment of social media posts.

1.3 Methodology

To sum our research proposal, we design a lexicon-based sentiment analysis approach for Arabizi by building a new Arabizi sentiment lexicon. We plan to create the sentiment lexicon in two phases (Generation and Expansion). We list the data requirements below.

1. Evaluation Dataset: A sentiment annotated dataset of Arabizi social media text to evaluate the lexicon in sentiment analysis.

2. External Lexical Resources: In the first phase of the lexicon, we generate new Arabizi sentiment words from other sentiment and DA lexical resources.

3. Arabizi Corpus: In the second phase, we expand the generated list of Arabizi sentiment words from a collection of social media Arabizi text using word embeddings.

We present this pipeline below in Figure 1.1. As can be seen, the research pipeline can be divided into two parts: Resources and Evaluation. We follow by detailing each of the presented steps.

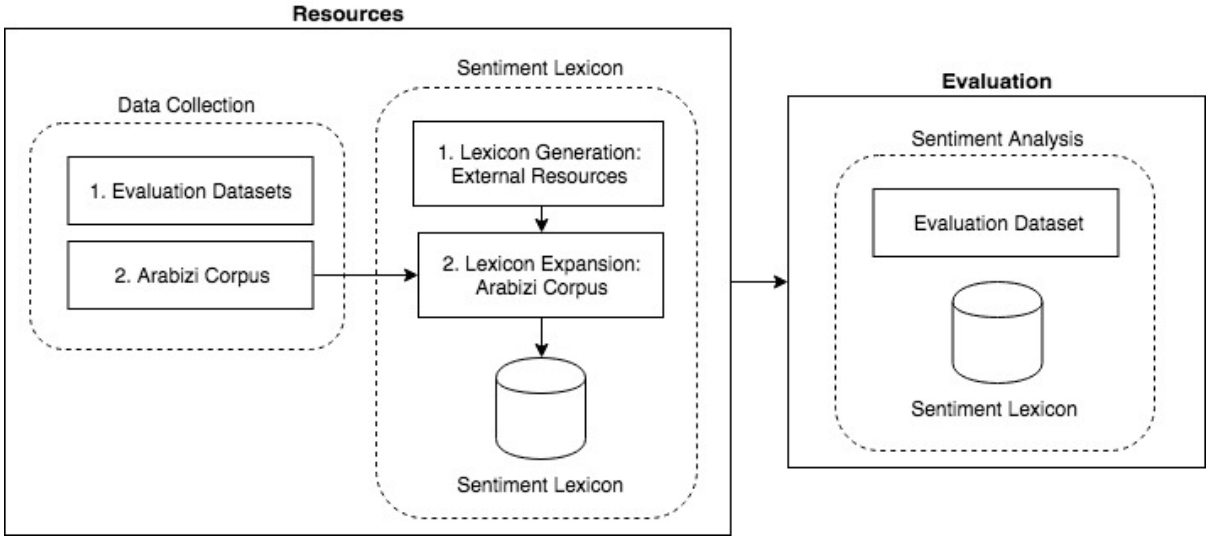


Figure 1.2 Methodology (Detailed)

Data Collection:

1. Evaluation Dataset: We create a social media evaluation dataset that is manually annotated with the help of three students. Each text in the dataset is annotated for language (Arabizi / Not Arabizi), and the Arabizi texts are annotated for sentiment polarity (Positive / Negative).

2. Arabizi Corpus: Since Arabizi is written in Latin script, we need to identify Arabizi texts from other Latin script languages. We use the language annotations of the evaluation dataset (Arabizi / Not Arabizi) to train a ML classifier to automatically identify Arabizi text from other Latin script languages in social media data.

Sentiment Lexicon:

1. Generation: The mentioned external resources go through a pipeline of translation, selection, and transliteration, also with the help of three students to generate a list of new Arabizi sentiment words.

2. Expansion: We expand the generated list of Arabizi sentiment words automatically to retrieve relevant forms for every sentiment word from the Arabizi corpus using word embeddings. We test different embedding models with different configurations.

Evaluation: We finally evaluate the generated list of Arabizi sentiment words and the resulting expanded sentiment lexicons for sentiment analysis against the created evaluation dataset to answer RQ2 and RQ3. We present this detailed pipeline in Figure 1.2.

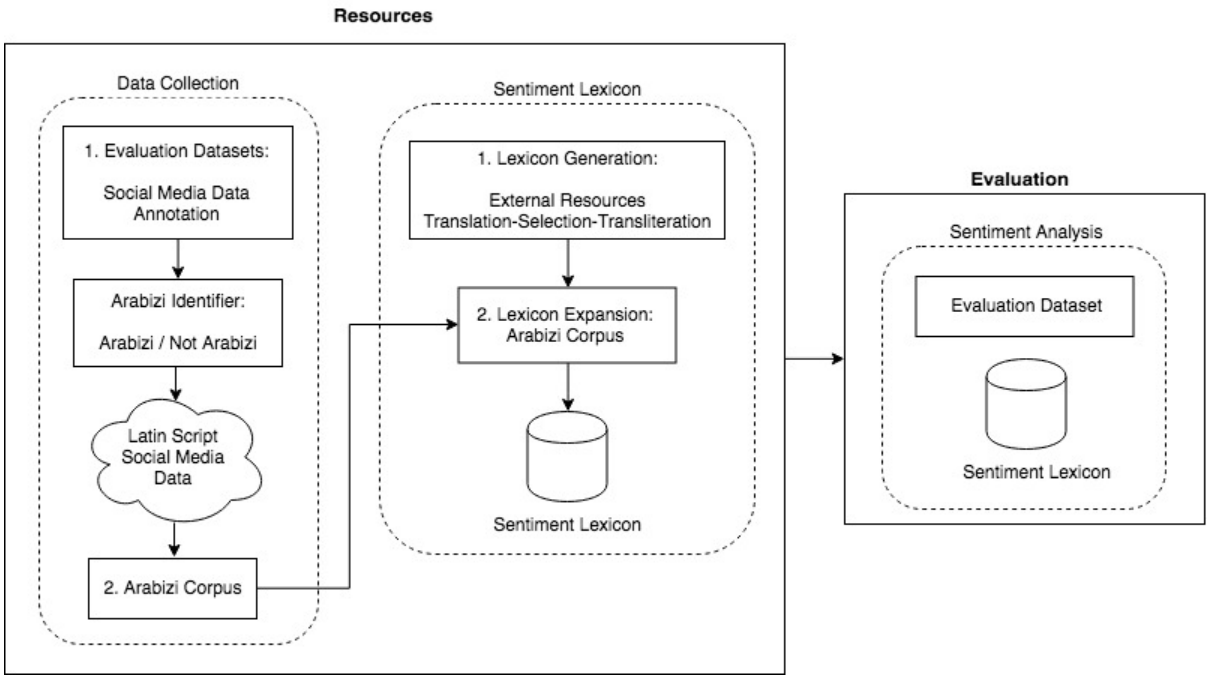


Figure 1.2 Methodology (Detailed)

1.4 Outline

We divide the thesis into four parts: Foundation, Resources, Evaluation, and Ending. We list and detail the chapters of each part below. Figure 1.3 presents the Resources and Evaluation parts.

Part I: Foundation

Chapter 2: *Background and Challenges*

In this chapter, we address RQ1; we initiate this thesis with a pilot study on the usage of Arabizi amongst other languages on Twitter across two different Arab regions. We then

present a fundamental linguistic background on Arabic and Arabizi, covering the orthography and morphology of Arabic and characteristics of Arabizi. We finally present the word classification challenges posed by these characteristics.

Chapter 3: *Literature Review*

In this chapter, we survey the literature of sentiment analysis in general and for Arabic in specific, highlighting different subtasks and popular approaches. We then narrow down to cover the related NLP work done for Arabizi. We discuss the strengths and weaknesses of the reviewed work and relate it to our proposed directions of research.

Part II: Resources

Chapter 4: *Data Collection*

In this chapter, we develop the necessary datasets to undertake the planned research. We describe the data collection and preparation to be used for the evaluation of the sentiment analysis approaches proposed in this thesis. We also compile an Arabizi corpus to be used for the lexicon expansion in Chapter 6.

Chapter 5: *SenZi: The Arabizi Sentiment Lexicon*

We build the sentiment lexicon in two phases. The first phase, Lexicon Generation, consists of a sequence of translation, transliteration, and selection of words from different resources. In this chapter, we detail these resources and the mentioned steps.

Chapter 6: *Lexicon Expansion*

In the second phase of the lexicon construction, Lexicon Expansion, we enrich the generated lexicon with relevant words automatically using word embeddings. In this chapter, we propose several expansions of the lexicon.

Part III: Sentiment Analysis

Chapter 7: *Evaluation*

In this chapter, we address RQ2 and RQ3 by evaluating the proposed sentiment lexicon and its expanded versions using the lexicon-based sentiment analysis approach against the prepared evaluation dataset. We follow the sentiment analysis experiments with an investigation of the classified data to reveal the advantages and limitations of the lexicon-based sentiment analysis for Arabizi. We finally discuss the potential research directions to address these limitations.

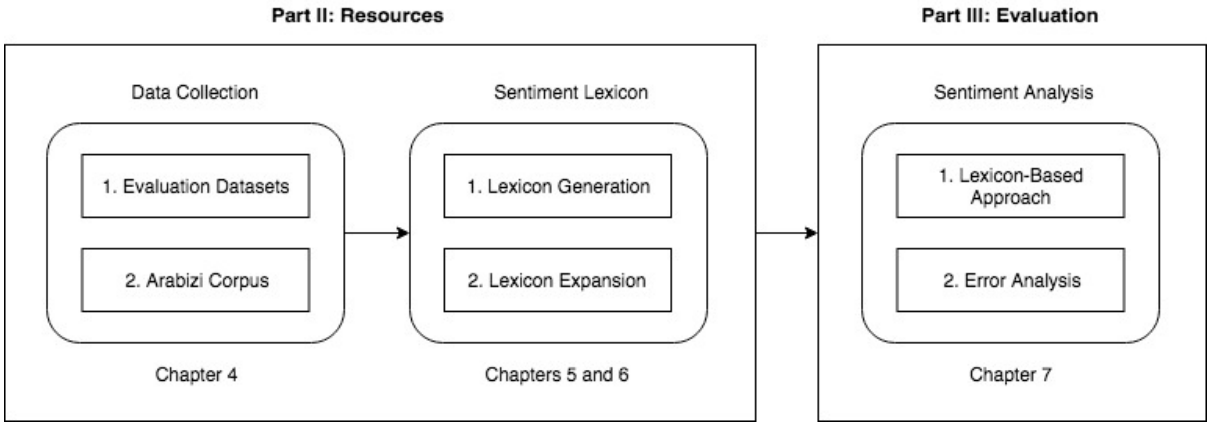


Figure 1.3 Resources and Evaluation (Outline)

Part IV: Ending

Chapter 8: *Conclusion*

We finally summarise our work, list and detail our contributions, present a future work plan, and draw some conclusions.

All publications and outcome resources can be found at the project's webpage:
https://tahatobaili.github.io/project-rbz/

I. Foundation

2 Background and Challenges

عد للحمى ودع الرسائل وعن الأحبة قف وسائل

واجعل خضوعك والتذلّل في طلابهم وسائل

والدمع من فرط البكاء عليهم جار وسائل

فاسأل مراحمهم فهنّ فهن لكل محروم وسائل

Omar bin Abul-Naseer

The term Arabizi is a portmanteau of *Araby* and *Englizi* which means Arabic and English. It is an unstructured Romanisation or Latinisation of the spoken dialectal Arabic (DA). In DA, new words and ways of speech are derived from the formal unified Arabic known as Modern Standard Arabic (MSA) or borrowed from an influencing foreign language or coined by the natives of the dialect. Arabizi reflects this non-standard spoken language in text.

Before the instant messaging technologies came into the Arab world, informal letters were either written in MSA or in another language. DA remained spoken among different Arab regions and there were no known intentions to make it a standard transcription. Instant messaging services started to become popular in the Arab world such as the Internet Relay Chat (IRC) and text messaging (SMS) but digital devices lacked an Arabic keyboard. Hence, bilingual Arabs worked-around this issue and used these services to communicate their DA using the Latin script keyboards which marked the birth of this phenomenon, now known as Arabizi (Yaghan, 2008). It has less popular names as well such as *Arablish* or *Franco-Arabic*.

Latinising Arabic not only made way to communicate Arabic in text but dialectal Arabic specifically. Some DA phonemes do not exist in MSA but sound close to English and French

phonemes. It therefore became easier to express these phonemes in Latin script than Arabic script. The Arabic letter ق for instance is pronounced as an emphasised *k* in MSA but *g* in most Gulf dialects, a voiced glottal stop in some Levantine and Egyptian dialects, and a soft *k* in one Palestinian dialect. Therefore, Arabizi users express their dialectal mother tongue in writing. The *khaliji* or peninsular Arabizi transcription of قلبي – *my heart* would be *galbi* as for the Lebanese and Egyptian it is *albi.* Similarly, the *Iraqi* or Mesopotamian *ch* phoneme of the letter ك - soft *k*; as in نحكي - *we-speak* sounds like *neHchi*. Transcribing such DA phonemes in the Arabic script used to be very unusual, therefore it seems that transcribing what is considered wrong in Arabic became socially acceptable to be transcribed in Latin script.

The Turkish language used to be written in Perso-Arabic script before 1923. After that it became Latinised with a unified orthography, for that they denoted special letters to represent the Turkish phonemes that had no equivalent in single Latin script letters such as Ç and Ş for emphasized and light *sh* phonemes respectively. Unlike Turkish, the Latinisation used for Arabic in Arabizi lacks a unified orthography rather it developed on its own. There has been no linguistic consensus on the orthography and Latinisatation style for social texting however numeral and compound letters to represent some consonant Arabic phonemes became the norm within communities but varies among regions.

Several studies pointed out that Arabizi users are young and more technologically fluent bilinguals usually between the age of 13 and 20 (Yaghan, 2008), (Allehaiby, 2013), (Al-Khatib & Sabbah, 2008). As such, any data analysis in the Arab regions might miss on relevant information from the youth if Arabizi is to be filtered from their datasets prior to the analysis.

Although computer mediated communication (CMC) gradually became Arabic friendly and prevalent in the Arab world, Arabizi is still widely used. (Muhammed, et al., 2011) studied the reasons for Arabizi usage to report that users find it easier and faster than typing in Arabic script. Some felt that Arabizi is a modern language that made them look cool or allows them to go with the flow. Others described themselves as relaxed as they type Arabizi because it is error-free and informal unlike MSA.

In this chapter, we address RQ1:

How frequently is Arabizi used on social media and what makes it a challenge for sentiment analysis?

We present a pilot study on the usage of Arabizi on social media. We then present a linguistic background on Arabic, briefing about its orthography, morphology, and phonology. We follow by describing the variant Latinisation of the Arabic phonemes. We finally present the challenges of the Latinised script for sentiment analysis.

2.1 Quantifying Arabizi in Social Media

In this section we analyse the languages used on Twitter across two Arab countries to highlight the percentage of Arabizi among other languages. Before presenting this study, we review related work on the percentage of Arabizi in private mediums such as the SMS.

(Muhammed, et al., 2011), (Yaghan, 2008), (Aboelezz, 2009), (Alabdulqader, et al., 2014), (GIBSON, 2015), (Jaran & Al-Haq, 2015), (Keong, et al., 2015) collected and analysed mobile chats and SMS data from selected participants summarised in Table 2.1.

Year	Location	Participants	Data	Size	Arabizi	English	Arabic
2015	Malaysia	20 Students	SMS	200	35%	50%	10%
2014	Egypt	26 Natives	SMS	~100K	77%	-	23%
2014	KSA	Natives	Mobile	~3K	15%	8%	74%
2012	Jordan		Forum	~460	35.5%	17.5%	32%
2008	Jordan	46 Students	SMS	181	37%	54%	9%

Table 2.1: Percentage of Arabizi Usage in Mobile Chats

Most of these studies reported that there is around 35% of Arabizi among English and Arabic messages from the mobile data of the students.

We now move on to analysing Arabizi messages in a public medium, Twitter. We focus this study on two Arab countries, Lebanon and Egypt.

2.1.1 Data Collection and Labelling

We used the Twitter streaming API[2] to collect live tweets, in 2016, that have geographic coordinates lying within the regions of Lebanon and Egypt. We extracted the language detected by the API, tweet location, country, and the language of the user from each tweet stream. For example:

ID	Tweet	Lang	Country	User ID	User Lang	User Country
001468231	لبنان ينتفض	AR	LB	4893812	EN	LB

Table 2.2: Example of a Tweet Stream

We collected two datasets, one from each country, and split into Arabic and Latin script tweets automatically. We present this distribution in Table 2.3.

Country	Tweets	Arabic	Latin Script
Lebanon	60.3K	47%	53%
Egypt	249K	70%	30%

Table 2.3: Arabic vs Latin Script Tweets

We randomly extracted a set of 5K tweets from each Latin script dataset and labelled it by language. However, Arabizi users often alternate between Arabizi and English within a single sentence; this is known as codeswitching. We labelled a tweet as Arabizi if the number of Arabizi words is higher than the number of English words. These words should consist of nouns and verbs not just connectors and stop words. For example:

*honestly allah y3afeke (recovery wish) that you still cant get over a story thats a year not my fault ur ex boyfriend was a *** sara7a (honestly)*

[2] https://developer.twitter.com/en/docs/tweets/filter-realtime/guides/connecting

The number of Arabizi words is lower than the number of English words
Label: Not Arabizi

kel marra b2oul monday bade ballesh diet bas emta ha yeje hayda lnhar
Everytime I plan to start a diet on monday but when will this day come

The number of Arabizi words is greater than the number of English words
Label: Arabizi

eh (yes) *God bless your mom w* (and) *your family*
Label: Not Arabizi

2.1.2 Results

After conducting the previously described labelling exercise our results show that, among the 5K randomly extracted tweets from Lebanon, 9.3% of the content is Arabizi, and from the 5K randomly extracted tweets from Egypt 19% is Arabizi. We present these results in Table 2.4.

Country	Latin Script Tweets	English	Arabizi	French	Other
Lebanon	5k	65%	9.3%	3%	22.7%
Egypt	5k	57%	19%	-	23%

Table 2.4: Distribution of Languages in the Latin Script Tweets

Interestingly we found that among the Latin script tweets in both countries there is around 23% of Latinised Far-Eastern languages. Far-Eastern expatriates living and working in the Arab region constitute a considerable portion of the population. These languages are mainly Filipino in Lebanon and Hindi in Egypt.

We removed these tweets from the analysis to present the distribution of Languages written by the natives of these countries. We re-calculated the overall percentage of Arabic, English, and Arabizi without considering the other Latin script tweets, presented in Table 2.5.

Country	Tweets	Arabic	English	Arabizi
Lebanon	60.3K	54%	40%	6%
Egypt	249K	75%	18.5%	6.5%

Table 2.5: Distribution of Languages from Natives of Lebanon and Egypt on Twitter

As can be seen from the results, Arabic dominates the languages in both countries however the percentage of Arabic to English differs greatly between Lebanon and Egypt with an almost equal percentage of 6% Arabizi.

2.1.3 Discussion

The results of this pilot study show that the percentage of Arabizi usage in Twitter data across both Lebanon and Egypt is lower than the findings by other researchers in mobile messaging, as shown previously in Table 2.1. We assume that people prefer to text in Arabizi on private mediums since it is generally perceived as an informal way of communication. However, 6% of a country's Twitter data reflects a considerable portion of the population's opinion. This motivated us to research this field and generate resources to process and analyse sentiment from Arabizi data.

2.2 Linguistic Background on Arabic

The way Arabs Latinise Arabic in text is based on the phonemes of their dialects and the orthography of Arabic; and since Arabizi reflects Arabic in Latin script, it naturally inherits its rich morphology. In this section we describe each of these factors for a better understanding of Arabizi.

2.2.1 Arabic Dialects

We describe some differences among Arabic dialects briefly and provide few examples.

Spoken or DA is categorised into the following major groups:

1. Peninsular: Yemeni, Omani, Qatari, Saudi, Emirati, Kuwaiti, and Bahraini.
2. Maghrebi: Moroccan, Tunisian, Libyan, and Algerian.
3. Sudanese: Chadian and Sudanese.
4. Egyptian
5. Levantine: Palestinian, Syrian, Jordanian, and Lebanese.
6. Mesopotamian: Iraqi.

We present an example of a positive phrase in different dialects in Tables 2.6 and 2.7 to show the difference in word choice.

Dialectal Variances of the Phrase

so pretty - (MSA) جميلة جداً

Arabic	Arabizi	Dialect
حلوة كتير	*7ilwe ktir*	Lebanese
حلوة مرة	*7ilwa marra*	Saudi
حلوة وايد	*7ilwa wayed*	Emirati
حلوة اوي	*7ilwa awi*	Egyptian
جميل برشا	*jmil barcha*	Tunisian

Table 2.6: Dialectal Variances (*so pretty*)

MSA	Gloss	Egyptian	Levant	North African
ذكي	Smart	lamma7, fahlawi, gamed	falteh, fo2is, 7arbou2	kafiz, 5afif, saji
أبله	Dumb	3abit, daye3, bati5a	mastoul, khales, ta2e2	mklej, mjmek, 7abes

Table 2.7: Dialectal Variances (smart and dumb)

Dialects evolve over time and change by influencing languages. Levantine and Egyptian for instance are influenced by Turkish. For example:

an interest - masla7a مصلحة

cynical - masla7ji مصلحجي

The common suffix *ji*, added to inflect *the-owner-of* or the *action-doer-of* a noun or a verb, originates from a Turkish morpheme *ci*.

Given that these dialects are spoken not written they pose a challenge for text analysis once transcribed in any script because there is no consensus on a standard transcription for any dialect.

Arabizi aims to convey the spoken DA words through text to the reader regardless of how it is written. For that, users spell words the way they sound in Arabic using the Latin script, however Arabic vowels are not consistently transcribed and there are several Arabic consonants that do not exist in Latin languages.

2.2.2 Orthography and Phonology

First of all, Arabic is written from right to left and each letter has an initial, medial, or final grapheme (shape) depending on its position in the word. Also, some letters connect with each other, others do not. The following examples show how each of these letters is written differently in isolation and connected in words.

ك ت ب ← كتب *he-wrote* أ ك ل ← أكل *he-ate* ط ر ق ← طرق *he-knocked*

Each Arabic grapheme represents a phoneme hence there is no need for compound letters to represent special phonemes as the *sh* and *th* in English. Although a letter might not be pronounced if compounded with another in specific contexts such as the ل *L* of the article ال - *al* in الشمس - *the-sun* written as *al-shams* but read as *ashams* (emphasised *sh*). These however, are pronunciation rules that depend on the combination of letters.

Arabic is nicknamed a throaty language for its guttural consonant letters. While the *v*, *p*, and *ch* Latin phonemes do not exist in MSA, there is the ح Ḥā', خ Khā', ع ʿayn, غ Ghayn, ق Qāf, and ء Hamza phonemes that are articulated in the post-velar areas of the oral cavity. Their phonetic description is listed in Table 2.8.

In simple terms, the ح Ḥā' sounds like soft H, خ Khā' is similar to the German *ch* in *Buch* or the Spanish *J* in *Juan*, ع ʿayn, as in the Arabic name *Omar*, غ Ghayn is similar to the French *r* in *Paris* or the Spanish *g* in *agua* but more emphasised, ق Qāf is a guttural emphasized *k* phoneme, and the ء named *hamza* is the stopping pronounced when a word begins with a vowel, such as the stop sound between the two words *an-apple*, to relate how a glottal stop can occur mid word in Arabic, لؤلؤة - *lu'lu'a* for example.

Arabic also contains light and heavy or stressed phoneme counterparts. طه - *taha* for example with an emphatic *t* ط is distinct from the light *t* ت. This special group of stressed consonants are ط Ṭā', ض Ḍād, ص Ṣād, ظ Ẓā', and ق Qāf sound like heavy *t, d, s, th* (as in there), and *k*. Their phonetic description is listed in Table 2.9.

Arabic is also unique in its phonetic vowel representations having short and long vowels. Though called long they sound slightly longer than the short ones. Long vowels ا ي و are alphabetic characters for ā y w. They are only three but the و wāw and ي yā' give different vowel phonemes based on the word. For example:

The و wāw in مجنون sounds like *ou*, *majnūn* while the و wāw in روان sounds like *w*, *Rawan*. The ي yā' in جميل sounds like a long vowel *i*, jamīl while the ي yā' in يمن sounds like *y*, *Yemen*.

Short vowels on the other hand are written as diacritics in formal MSA. Diacritics are marks that go above or under the letters such as:

ك ت ب: كَتَبَ - *kataba* كُتُبْ - *kutub* كَتَّبَ - *kattaba*

Arabic Letter	Name	Phonetic Description
ح	Ḥā'	Voiceless pharyngeal constricted fricative
خ	Khā'	Voiceless velar fricative
ع	ʿayn	Voiced pharyngeal fricative
غ	Ghayn	Voiced velar fricative
ق	Qāf	Voiced uvular plosive
ء	Hamza	Voiceless glottal stop

Table 2.8: Arabic Guttural Consonants

Arabic Letter	Name	Phonetic Description
ط	T̲ā'	Emphatic voiceless dental plosive
ض	D̲ād	Emphatic voice alveolar plosive
ص	Ṣād	Emphatic voiceless alveolar fricative
ظ	Ẓā'	Emphatic voiced alveolar fricative
ق	Q̲āf	Voiced uvular plosive

Table 2.9: Arabic Heavy Consonants

These are different words with different meanings (*he wrote, books,* and *he made someone write*). The diacritics, or vowels, are placed based on the grammar and part of speech. There is a diacritic for no-vowel *sukun* and a diacritic for emphasis *shaddah* gemination which denotes a double letter phoneme such as the double *m* in *Muhammad*. An example where the the gemination changes the meaning of the word:

ع ذ ب: عذب - *sweet (adj. for drink)* ع ذ + *gemination* + ب: عذّب - *torture*

As integral the diacritics are to the language, most of the times they are not written in everyday text especially in digital format because Arab natives with basic MSA education would know how to read a non-diacritcised text even if they were not so accurate they would infer the meaning of the words from their context. طه *taha* is a two-letter name, the emphatic *t* ط and the *h* ه only, the vowels are pronounced naturally.

In Section 2.3 we show examples of how Arabizi users map these distinctive Arabic phonemes in Latin script.

2.2.3 Morphology

In this section we describe some of the Arabic's rich morphology and the complexity of stemming morphologically shifted words computationally.

Morphemes in Arabic signify the relationship between nouns and verbs. Arabic is rich in morphology because these morphemes take place by a change or extension in the root forms of the nouns and verbs.

وستبشرونها *wasatubashirūnahā* for example, means *and-you-will-inform-her* derived from the verb بشر meaning *to inform about something delightful*. We break it down for clarity, the underlined word is the root verb, the rest are clitics and pronouns.

و + س + ت + بشر + ون + ها

wa sa tu bashir ūna hā

and + will + you 2nd person case + inform + plural you pronoun + her

Most root words in Arabic are triliteral, consisting of 3 letters, from which words are derived. There are two layers of morphology from the root words in Arabic, derivational morphology and inflectional morphology. Considering the root to be the lexeme, the unit of meaning, then the first layer of forms are the lemmas which are words derived from the root. For example, the lemmas لعبة *toy*, لاعب *player*, and ملعب *playground* or *stadium*, are all derivations from the root word لعب *play*. The second layer of forms are inflections of the lemmas or the root such as العاب *toys*, لاعبون *players*, and ملاعب *playgrounds* (inflections of the mentioned lemmas) and نلعب *we-play*, يلعب *he-plays*, ألعب *I-play* (inflections of the root word لعب *play*). This is presented in Figure 2.1.

Figure 2.1 An Example of Arabic Morphology

For consistency, we will call the word forms that are derived from the root such as the ones from the first layer, lemmas, and word forms that are inflections of the lemmas or the root such as the ones from the second layer, inflections, throughout this thesis.

This morphology builds upon the root word by a change in the diacritics or an addition of an affix (prefixes, suffixes, or infixes) or a replacement of letters or a dropping off some letters, or an adhering of prepositions (clitics).

1. Diacritics: Different diacritics infer different meanings as mentioned in Section 2.2.2. Changing a diacritic generates a new inflection, the following examples show different diacritics for the same word:

   لعبتْ *she played* لعبتِ *you* (feminine) *played* لعبتَ *you* (masculine) *played* لعبتُ *I played*

2. Affixes: Prefixes and suffixes are the morpheme units added before or after the root or lemma:

   م + لعب: ملعب prefix + *play* = *playground*

   لعب + ن: لعبن *play* + suffix = *they* (feminine) *played*

   ي + لعب + ن: يلعبن prefix + *play* + suffix = *they* (feminine) *are playing*

   In addition to the prefixes and suffixes, Arabic morphology includes infixes. Infixes are morpheme units added within the roots or lemmas.

   لعب *play* ل + ا + عب: لاعب *player*

   ملعب *playground* مل + ا + عب: ملاعب *playgrounds*

   ت + ل + ا + عب: تلاعب noun prefix + (*play* with infix) = *messing* or *playing around*

3. Replacement and Dropping off Letters: Sometimes a vowel letter drops from the root.

   وزن *weight*: زان *he measured the weight* زن: (imperative) *weigh*

The و *wāw* vowel is dropped The و *wāw* vowel is replaced by the ا ā vowel

4. Clitics: Proclitics and enclitics are prepositions attached to nouns or verbs, before or after the word to become a single word.

preposition (*and/with*) + played = *and he played* ولعب :و + لعب

This is very frequent in DA with more adhering prepositions.

preposition (*on top of*) + *my head* = *on my head* عراسي :على + راسي

A common expression meaning *with pleasure*

We now show the difficulty of extracting the root of the word automatically from an inflection.

A stem is defined as the part of the word that remains after removing all affixes and clitics, the root word in this case. Stemming is the automatic extraction of the stem from words. If it is possible to stem the words, then all of the inflectional and orthographical forms would be easily mapped with their lemmas. However, since most Arabic roots are triliteral it is difficult to determine which 3 letters in a word makes the root. For example, two opposite sentiment words can be extracted from the word نكرم *we-be generous*:

نكرم: نكر *to deny* نكرم: كرم *generosity*

Or the root word from the first example *wa-sa-tu-bashir-ūna-hā:*

بشر: *inform, delight* بشرّ: with evil

Also, the root could be lost since letters might drop after a morphological shift or an adhering of a preposition.

تتصل: *to connect/call* Root :وصل

The *wāw* vowel is dropped

وعيد: Root: *promise* وعد

و + عيد: preposition + *holiday: and holiday* وعيد

Some works in the literature attempt to stem Arabic, however these works are designed for MSA only and rely on dictionaries. ElixirFM[3], a morphological analyser is based on the Prague Arabic Dependency Treebank by (Smrž, 2007). $CALIMA_{star}$[4] also depends on a database of predefined tables (Taji, et al., 2018). Other papers proposed to process words and measure the similarity with a predefined list of MSA roots such as (De Roeck & Al-Fares, 2000). (Taghva, et al., 2005) on the other hand, developed a rule-based stemmer. As can be seen the complexity of stemming Arabic is very challenging that researchers had to build databases to retrieve MSA stems from inflected words. This type of stored data or rule-based stemmers is most likely very limited that replicating it for different Arabic varieties requires rebuilding it from scratch.

2.3 Characteristics of Arabizi

2.3.1 Transcription

Bilingual Arabs found a way to represent the mentioned guttural and heavy consonants in Latin script by using either numeral or compound letters. Those representations became normalised differently in every Arab region (Aboelezz, 2009), (Allehaiby, 2013), (BIANCHI, 2012), (Duwairi, et al., 2016) studied these normalisations in Egyptian and Jordanian Arabizi. (Sullivan, 2017) studied the normalised Arabizi in Lebanon. Most of these normalisations are presented in Table 2.10.

Some representations are based on graphemes, shapes of the letters, and some on phoneme similarity, for example the numerals 3 and 7 to represent the ع and ح are chosen based on the grapheme similarity however the compound letters *kh* and *gh* to represent the خ and غ are based on the phoneme similarity.

[3] http://quest.ms.mff.cuni.cz/elixir/

[4] https://calimastar.abudhabi.nyu.edu/#/analyzer

	Arabic Letter	Name	Arabizi: Egypt and Jordan	Arabizi: Lebanon
Guttural Consonants	ح	Ḥā'	7 / h	7 / h
	خ	Khā'	7’ / 5 / kh	kh / 5
	ع	ʿayn	3	3
	غ	Ghayn	3’ / gh	gh / 8
	ق	Qāf	8 / 2	2
	ء	Hamzah	2 / ‘	2
Heavy Consonants	ط	Tā'	6 / t	t
	ض	Dād	9’ / d	d
	ص	Ṣād	9 / s	s
	ظ	Ẓā'	6’ / z / th	z
	ق	Qāf	2 / ‘	2

Table 2.10: Arabizi Representations

Although the consonant letter representations have been normalised, as can be seen from the table this normalisation is inconsistent. Some guttural consonants are represented by two or more Latin alpha numerals.

حبيبي Ḥābībī - *my darling: 7abibi* or *habibi*.
7 or *h* to represent the ح Ḥā phoneme.

On the other end, the Arabizi representations for the heavy consonants are the same for their light consonant counter parts.

Both *lesson* - *dars* درس and *tooth* - *Dars* ضرس are written as *dars*.

However, the style of Latinising the Arabic vowel phonemes has not been normalised. The Latinisation of vowel letters is inconsistent because transcription of vowel letters is optional and each user interprets how vowel letters should be represented on their own. First, there is no burden in transcribing vowel letters as the text is readable and comprehensible without vowel letters. Therefore, users might transcribe or opt out from transcribing the vowel letters, or transcribe them intermittently even within the same word. For example:

حبيبي Ḥābībī: habibi, habb, hbb, 7abibi, 7abebe, 7bb, 7abb, 7abeeb, 7abibeh

Second, transcription of vowel letters depends on the dialect of the users and their own perception of spelling vowels in English.

خير *khāyr* - *good* is pronounced as *khāyr* in one Lebanese dialect and *kher* in another Lebanese dialect. It is therefore quiet common to be transcribed as *kher* or *khayr* in Arabizi. However, one's perception of spelling this *āy* vowel could be *ei*, thus *kheir* is also common.

خير Khāyr: kher, kheir, khayr, khyr.

Although majority of the consonant letters have been normalized they still present challenges for deciphering the text and the non-unified Latinisation of vowel letters gives a range of possibilities to transcribe Arabizi, for that Arabizi is free from language policing, it is social, informal, relaxed, fast and fun but poses several challenges for processing.

2.3.2 Codeswitching

Switching between Arabizi and Latin script languages intermittently is very common for Arabizi users. The pilot study presented earlier in this chapter shows that English is the dominant codeswitching language with Arabizi in Lebanon and Egypt. Codeswitching may occur either inter-sentential or intra-sentential, that is within individual sentences or within conversations. We present some examples from social media.

Tweet: *bonsoir 7ewalit a3melik add 3ala fb bass i didnt find you can you give me your account.*
Languages: French, Arabizi, and English

In Figure 2.2 we present a snapshot of a Facebook post where Arabizi and English are codeswitched within the same sentences. In Figure 2.3 we present a snapshot from a Facebook page, where Arabizi, English, and Arabic occurs in a single conversation.

Borrowing is also common in DA and has been reflected in Arabizi where a word from different language is borrowed and integrated with the DA morphology.

Love you: *luv* + *ik* = *luvik* (feminine) and

miss you: *miss* + *ak* = *missak* (masculine)

In the next section, we present the challenges posed by the mentioned characteristics for sentiment analysis.

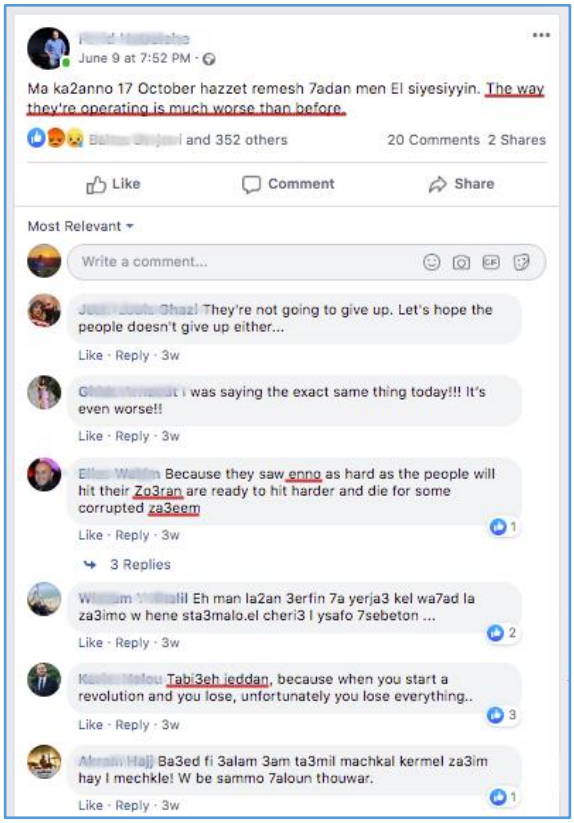


Figure 2.2 Intra sentential codeswitching

Figure 2.3 Inter sentential codeswitching

2.4 Arabizi Challenges for Sentiment Analysis

In this section we explain how the mentioned characteristics of Arabizi introduce challenges and limitations for word classification and transliteration.

2.4.1 Creating Datasets

Any sentiment analysis approach we consider for this research requires an evaluation dataset to measure the value of the proposed approach.

As can be seen from the results of the pilot study in Section 2.1, Arabizi consists of around 6% of the Twitter data in Lebanon and Egypt, which is 13% of Lebanon's and 26% of Egypt's Latin script tweets. Since Arabizi is low in resources we need to create a new Arabizi dataset and annotate it for evaluating the sentiment analysis approach. However, the nature of the Arabizi script poses a challenge in collecting the dataset because we need to identify Arabizi from English as a first step before annotating the data with sentiment labels.

If Arabizi comprises 13% of Lebanon's Latin script tweets, it then requires a costly annotation of 10K tweets to generate a dataset of 1.3K Arabizi tweets. Such annotation has to be carried out carefully as well in the light of codeswitching.

2.4.2 Word Ambiguity

A word is considered ambiguous for classification if it has several meanings or connotations. Although homonymy[5] is natural among languages. Transcribing Arabic phonemes that do not exist in the English Latin script causes an additional word ambiguity.

Ambiguous words in Arabizi are formed by transcribing a short Arabic vowel phoneme (a diacritic) as a vowel letter in Latin script (vowel ambiguity) or transcribing one Latin script letter for two distinct Arabic letters such as the soft and heavy consonants (consonant ambiguity). This harms sentiment classification if a neutral word is ambiguous for a positive or a negative word. We present some Lebanese dialect examples below:

1. Vowel Ambiguity:

   village - ضيعة as *day3a* (short vowel /a/ a diacritic originally ضَ)
   confused or *lost* - ضايعة as *day3a* (long vowel ā ضا)

   stupid - غبي as *ghabe* (short vowel /a/ a diacritic originally غَ)
   forest - غابة as *ghabe* (long vowel ā غا)

2. Consonant Ambiguity:

   route - درب as *dareb* (soft *d* د)
   hit or *harm* - ضرب as *dareb* (heavy *d* ض)

[5] The relation between words with identical forms but different meanings.

Word ambiguity also impacts the task of transliteration. Transliteration in this scope is the automatic conversion of words written in Latin script, Arabizi, into the Arabic script, Arabic, in other words, de-Latinisation.

Online transliterators have been developed for several Latinised languages e.g. Chinese, Hindi, and Arabic. The purpose of the Arabic transliterator is for users to type in Latin script at their comfort and receive output text in Arabic script, however, given the limited consonant phonemes and the inconsistent choice of vowel letters in Arabizi, such transliterators disambiguate words by generating a list of possible transliterations for every typed word. Microsoft[6] and Google[7] released online transliterators, Yamli[8] however is one of the most popular Arabic transliterators, having lived for longer than Microsoft and Google. We present snapshots of Yamli's suggested transliterations for the ambiguous words mentioned in the examples earlier in Figures 2.4, 2.5, and 2.6.

1. Vowel Ambiguity:

Lost / *Village*

Figure 2.4: Transliteration Example 1

Stupid / *Forest*

[6] https://www.microsoft.com/en-gb/download/details.aspx?id=20530
[7] https://www.google.com/inputtools/try/
[8] https://www.yamli.com

Figure 2.5: Transliteration Example 2

2. Consonant Ambiguity:

Route / *Hit*

Figure 2.6: Transliteration Example 3

As such, the word ambiguity formed by the inconsistent Latinisation of Arabic makes the task of transliterating whole Arabizi datasets simply infeasible with its current state. In Chapter 3 we review some papers that attempt to automate the transliteration of Arabizi.

2.4.3 Sparsity

Coverage is the major challenge in the lexicon-based sentiment analysis approach. Arabic is rich in morphology that some sentiment words may have over a hundred inflections. This is even juxtaposed with a transcription that lacks a unified orthography resulting in a large number of inflectional and orthographical variants for each word.

In Section 2.2 we showed the layers of Arabic morphology where lemmas derive from triliteral root words and inflections derive from lemmas or from the roots directly. We now present some of this structure for the sentimental word حبّ *7obb - love* in Lebanese dialect Arabizi, lemmas in Table 2.11 and inflections in Table 2.12.

ma7boub	*Beloved*
ma7abbeh	*Affection*
mu77ib	*Loving*
mu7abab	*Lovable*
ta7abob	*Endearment*
mt7abeb	*Endearing oneself*
7abib	*Lover*
t7bib	*Make desirable*
musta7ab	*Preferable*
sta7ab	*Appreciate*
ta7ab	*Mutual love*
muta7ab	*Amicable*
mu7abaz	*In favour of*

Table 2.11: Lemmas of the word *7obb - love*

	Present	Past
I love	*b7ib*	*7abeit*
I love you (singular and plural)	*b7ibak, b7ibik, b7ibkon, 7abibi, 7abibti, 7abibete*	*7abeitak, 7abeitek, 7abeitkon*
I love him, her	*b7ibo, b7iba*	*7abeito, 7abeita*

I love them	*b7ibon, b7ibhon*	*7abeiton, 7abeithon*
You love	*Bet7ib, bet7ibe*	*7abet, 7abeite*
You love him, her	*Bet7ibo, bet7eba, bet7ibi, bet7ibiya*	*7abeito, 7abeita*
You love them	*Bet7ibon, bet7ebiyon*	*7abeiton, 7abaiteyon*
He/she loves	*Be7ib, bet7ib* *Y7eb, t7eb*	*7ab, 7abit*
He/she loves you (singular & plural)	*be7ibak, bet7ibak, be7ibkon, bet7ebkon, y7ebak, y7ebkon, t7ebak, t7ebkon*	*7abak, 7abitak, 7abkon, 7abitkon*
He/she loves him/her	*be7ibo, bet7ibo, be7iba, bet7iba, y7ebo, y7eba, t7ebo, t7eba*	*7abo, 7abito, 7aba, 7abita*
He/she loves them	*be7ibon, bet7ibon, y7ebon, t7ebon*	*7abon, 7abeton*
We love	*men7ib, n7ib*	*7abeina*
We love you	*men7ibak, men7ibek, men7ibkon, n7ebak, n7ebik, n7ebkon*	*7abeinek, 7abeineke, 7abeinekon*
We love him, her	*men7ibo, men7iba, n7ibo, n7iba*	*7abeineh, 7abeineha*
We love them	*men7ibon, n7ebon*	*7abeinehon*

Table 2.12: 90 Lebanese Dialect Inflections for the word *7obb love*

The mentioned issues of inconsistent orthography and richness in morphology lead to a high degree of lexical sparsity. Creating a sentiment lexicon with one or few forms for each positive and negative word is unlikely to be sufficient to cover the large number of possible variants for each of these sentiment words.

As such, the very large magnitude of lexical sparsity by Arabizi defies the fundamental technique of sentiment analysis which is classifying words, the challenging question is hence: *How can we create a lexicon of sentiment words with all its forms?*

This large magnitude of lexical sparsity is also challenging for the machine learning approach for sentiment analysis which is learning the sentiment from the composition of words, the challenging question hence becomes: *how large the labelled datasets should be to cover all the forms of sentiment words?*

Anticipating the complexity of the lexical sparsity challenge for both approaches, we decided to induce orthographically and morphologically rich sentiment lexicons for Arabizi as automatic as possible.

Codeswitching also impacts transliteration and sentiment analysis especially if English sentiment words overlap in the spelling with Arabizi words of opposite sentiment. The word *kiss* for example would transliterate to a widely used vulgar swearing word in Arabic.

2.5 Chapter Summary

In this chapter we addressed RQ1 by presenting a pilot study on the usage of Arabizi on Twitter and describing the characteristics of Arabizi that introduced challenges for sentiment analysis.

We found that Arabizi constitutes of around 6% of Lebanon's and Egypt's Twitter data. We then explained some of the differences among dialects and how these dialects are reflected in Arabizi texts. We provided a linguistic background on the phonology, morphology, and orthography of Arabic. We finally presented some of the transcription styles of Arabizi and how it generates word ambiguity and high degree of lexical sparsity that defy NLP tasks such as sentiment classification and transliteration.

3 Literature Review

ألمٌ ألمٌ ألمٌ ألمٌ بدائله إن أنّ أنّ أنٌّ أنّ أنّ

Mutanabbi

In this thesis we investigate the application of a popular NLP task, sentiment analysis, onto a new domain, Arabizi, a variety of Arabic. Therefore, we divide this chapter into three sections: Sentiment Analysis, Sentiment Analysis for Arabic, and Arabizi in NLP.

In the first section we give a general introduction about sentiment analysis covering the sub-tasks and some advancements. In the second section we survey sentiment analysis for Arabic covering the lexicon-based and Machine Learning (ML) approaches. In the third section, we detail what researchers have done for Arabizi in the scope of NLP. We end each section with a short discussion about the strengths and limitations of the reviewed work. By the end of the chapter we discuss how our work relates to and differs from that of the reviewed literature.

3.1 Sentiment Analysis

3.1.1 Overview

(Liu, 2015) defined three types of sentiment analysis: Document Level, sentence level, and Aspect level. Document level focuses on the overall opinion of a document. Sentence level classifies individual sentences into positive, negative, or neutral. The more fine-grained sentiment analysis type is the aspect level that extracts opinion towards targets found in text.

(Cambria, et al., 2017) divided the task of sentiment analysis into three layers: Syntactics layer, semantics Layer, and pragmatics Layer. Each layer focuses on subtasks to reach an ideal sentiment classification of texts. We brief each layer below.

The syntactics layer deals with understanding the grammar of the text. It focuses on simplifying the text to reach a readable format. An example subtask of this layer is Lemmatization, which aims to reduce inflected words to their base form, mentioned in Chapter 2.

The semantics layer deals with understanding the literal meaning of the text. It focuses on extracting concepts from the simplified text such as detecting named entities (*person, organisation, location*) and identifying subjective text. Classifying a text as subjective or objective is a task known to precede sentiment analysis called Subjectivity Detection (Liu, 2015).

The pragmatics layer deals with understanding what the text is trying to convey. It focuses on extracting meanings from the text which includes sarcasm detection, aspect extraction, and polarity classification.

Aspects are the opinion targets for example:

my phone is great but the battery life is poor
Phone and *battery life* are the aspects for the opinions *great* and *poor*.

Polarity classification is the heart of sentiment analysis. It is the task of classifying a given text as positive or negative. It is the main focus of our research in this thesis. We even refer to the term sentiment analysis for polarity classification.

Our evaluation dataset is Arabizi social media text, Twitter data in specific (Chapter 4). We evaluate the effectiveness of using the proposed lexicon in classifying an annotated set of tweets (positive, negative). We do not segregate the annotated tweets into sentences, rather we classify the whole piece of text in each tweet. As such, we consider this type of sentiment analysis: Polarity classification for short documents.

We now present the different approaches used in the literature of sentiment analysis. Some of this information comes from two recent survey papers (Yue, et al., 2018) and (Zhang, et al.,

2018) that rendered an extensive work in reporting the advances of sentiment analysis in the literature. We also mention some of the seminal works in sentiment analysis.

Many of the mentioned works use data-driven approaches that depend on data that have been prepared at an earlier time and was ready to use. Although we did not study a Machine or Deep Learning (DL) approach for sentiment analysis in this thesis, mainly because Arabizi is very low in data resources and creating training data for ML or DL is very expensive, in terms of time and price, for the large size of data required to cover sufficient vocabulary given the high degree of lexical sparsity, we present the following works to provide a background on the state of the art of sentiment analysis and how it developed.

3.1.2 Supervised Machine Learning vs. Lexicon-based Approaches

(Pang, et al., 2008) explored the effectiveness of applying the supervised ML algorithms Naïve Bayes (NB) and Support Vector Machines (SVM) to the sentiment classification of movie reviews. These algorithms are called supervised because they depend on training a robust sentiment classifier from manually labeled data (Hu, et al., 2013), known as machine learning approaches. As such, ML approaches require manual labeling of data particularly if the language lacks dataset resources (Pak & Paroubek, 2010) (Barbosa & Feng, 2010), (Kouloumpis, et al., 2011).

(Barbosa & Feng, 2010) used sources of noisy labels as a training dataset instead of annotating data for sentiment classification. They studied the effect of different combinations of these features. They used meta-information associated with the words such as the characteristics of how the text is written.

Unsupervised sentiment analysis on the other hand is the task of classifying text without the need for a labelled dataset. A classical application to this is the lexicon based (LB) approach where a given lexicon determines the polarity of the words in a sentence leading to the overall polarity of the sentence (Thelwall, et al., 2012), (O'Connor, et al., 2010) (Bollen, et al., 2011).

(Turney, 2002) also used an unsupervised method for detecting polarity of products and movie reviews. They check the pointwise mutual information (PMI)[9] between a given phrase and the word *excellent* minus the PMI between that phrase and the word *poor*.

SentiStrength (Thelwall, et al., 2012) and SentiWordNet (Esuli & Sebastiani, 2007) are publicly large English sentiment lexicon that have been used in sentiment analysis research and applications frequently such as (Tellez, et al., 2017) who proposed a sentiment analysis and polarity classification framework that relies on the part of speech (POS) information found in SentiWordNet.

(Saif, et al., 2014) proposed SentiCircles, a lexicon-based approach that builds a dynamic representation of context to tune a pre-assigned strength and polarity of words found in a lexicon. They incorporated the contextual and the conceptual semantics of the words.

Lexicon-based approaches might suffer from low recall values because they are limited to the words that comprise the lexicon to determine the orientation of opinion or sentiment not coping with the neologism of the social media. ML approaches on the other hand depend on annotated data, a serious challenge in the scope of low-resourced NLP.

(Zhang, et al., 2011) combined the unsupervised with a supervised approach. They started with a Lexicon-based approach to label tweets using a publicly available sentiment lexicon. They extracted sentiment cues from the automatically labeled dataset using Chi-square test[10]. Afterwards, they used the labelled dataset to train a ML SVM sentiment classifier. They key to this approach is the good accuracy of the lexicon based approach, otherwise the training data fed by the ML sentiment classifier would be falsely labelled.

3.1.3 Deep Learning in Sentiment Analysis

Deep learning has emerged as a powerful machine learning technique that learns multiple layers of representations or features of data and produces state of the art prediction results

[9] https://en.wikipedia.org/wiki/Pointwise_mutual_information
[10] https://en.wikipedia.org/wiki/Chi-squared_test

(Zhang, et al., 2018). This section will list and explain briefly some of the common deep learning algorithms then reviews some papers that applied these algorithms to sentiment analysis.

DL uses a cascade of multiple layers of nonlinear processing units for feature extraction and transformation. These layers are called neurons in an artificial neural network (NN). A NN is a complex of input, output, and some hidden layers. Connections between neurons are associated with values that control the signals or the inputs that come out as outputs from neurons and go in the following layers as inputs. After training a NN it will generate a hypothesis out of the data.

The recurrent neural network (RNN) have directed cycles back to its neurons that leverage the network to remember processed information. Bidirectional RNN consists of two RNNs that are stacked on top of each other. Long Short Term Memory (LSTM) consists of four NN layers that are capable of learning long-term dependencies. Recursive neural network (RecNN) learns a tree structure from input sentences in a bottom-up fashion to generate phrase representations.

(Socher, et al., 2011) used a recursive auto-encoder to learn representations of multi-word phrases for sentiment analysis over an online dataset of public's reactions to people's confessions. Then in another work, they presented a Matrix-Vector Recursive Neural Network that learns the meaning vectors of a word and how that word modifies its neighbors (Socher, et al., 2012). In (Socher, et al., 2013) they proposed the Recursive Neural Tensor Network model (RNTN) which computes compositional vector representations for phrases of variable length. These representations were then used as features to classify each phrase. (Santos, et al., 2015) trained a deep NN on character, word, and sentence level representations showing that this approach is as affective as the RNTN approach for sentiment analysis.

A great attention has been given to word embeddings recently for its capabilities in NLP. Word embeddings are NN based models that are known for bilingual lexicon induction (BLI), but also they are being used in sentiment analysis. A word embedding space is a vector space of word representations generated from a large corpus by converting the

vocabulary of the corpus into vectors of real numbers. Each dimension of the vector represents a latent feature or a linguistic pattern.

(Tang, et al., 2016) proposed learning sentiment specific word embedding for sentiment analysis. They encoded the sentiment into the continuous vector representation of words to separate words of opposite sentiments. They trained the sentiment specific word embedding from tweets and developed three NNs to incorporate the supervision from sentiment polarity of text in their loss functions. (Wang & Xia, 2017) applied a similar approach as well. (Vo & Zhang, 2015) proposed contextual representation for target Twitter sentiment analysis. They incorporated sentiment lexicon information and distributed word representations. (Zhou, et al., 2015) trained a bilingual sentiment word embeddings for English and Chinese. They incorporated sentiment polarities of text into the bilingual embeddings by employing a labeled corpora and their translation.

In recent years the science of NLP was boosted by the release of NN models that learn from large compilations of text and can be later fine tuned for downstream tasks such as sentiment analysis, namely ELMO (Peters, et al., 2018), Ulm-Fit (Howard & Ruder, 2018), and more recently BERT (Devlin, et al., 2018) outperforming the state of the art in several NLP tasks. BERT is built using a bi-directional transformer. The transformer is a NN architecture that consists of encoding and decoding layers that gives attention to the input parts that are most relevant. BERT has been trained on 104 languages including Arabic but not Arabizi.

3.1.4 Discussion

Taking into consideration the scarcity of the required sentiment-annotated datasets to train an Arabizi ML sentiment analysis approach and the cost to develop such datasets (Chapter 2), in this thesis we design a lexicon-based approach as our study case for Arabizi sentiment analysis. Although developing a new sentiment lexicon is not a simple task, we explore the power of a word embeddings to partially automate the creation of the proposed sentiment lexicon.

As can be seen from the mentioned works, deep learning for sentiment analysis have developed from a state of well-established datasets such as the LSTM and the RNN. As for

the word embeddings, the latter approaches proposed enriching the embedding with sentiment information of words for a polarity representation of the embedding space. In this thesis we propose to use word embeddings the other way around, we enrich an immature sentiment lexicon from the word embedding representations to build it.

3.2 Sentiment Analysis for Arabic

In this section we study some of the most related works in the literature of sentiment analysis for Arabic. We show where this field has come to and discuss its limitations for Arabizi.

Arabic falls behind English in NLP because it is lower in resources and considered more challenging for its script, varieties, and morphology. We review efforts for building lexical resources and applying ML techniques for sentiment analysis.

As shown in Chapter 2, many DA words differ from MSA to a great extent. Since Arabizi is a variety of DA, we focus the review solely on the approaches; the results on MSA do not serve our work a great purpose. We relied on (Al-Ayyoub, et al., 2019) survey paper for reviewing the main works below.

3.2.1 Lexicon Based Approaches

(Elhawary & Elfeky, 2010) created an Arabic weighted sentiment lexicon by taking a set of labelled phrases and used Arabic word similarity graphs with the labelled phrases. (Farra, et al., 2010) also used a lexicon-based approach but taking the frequency of words and sentence structure into account.

Sifaat (Abdul-Mageed & Diab, 2012), an Arabic lexicon built from 3.3K sentiment-labelled adjectives and expanded into 229K words by translating three English lexicons using Google Translate.

Tharwa (Diab, et al., 2014), a large-scale Arabic lexicon containing parallel words from MSA, Egyptian dialect Arabic, and English. They compiled previous Egyptian Arabic lexical resources. They maximised the number of Egyptian dialect variants to 73K words. Then they

manually mapped these words with MSA and English equivalents along with their POS tags. Finally, they evaluated the lexicon manually, with the help of annotators, and automatically using multilingual parallel corpora.

SANA (Abdul-Mageed & Diab, 2014), a large-scale multi-dialect sentiment lexicon for Arabic. They compiled 44K positive, 49K negative, and 132K neutral words using existing Arabic lexicons, SIFAAT and HUDA. Then they translated the English SentiWordNet (Esuli & Sebastiani, 2007), a Youtube Lexicon, and the Affect Control Theory Lexicon to Arabic and mapped SentiWordNet with the mentioned Tharwa lexicon. They used the PMI (Turney, 2002) of positive and negative terms of Twitter and Yahoo Maktoob[11] datasets. They finally evaluated the lexicon by annotating random sets and measuring the polarity agreement among the lexicons.

(Alhazmi, et al., 2013) created Arabic SentiWordNet (ASWN) using the English SentiWordNet (ESWN) 3.0 (Baccianella, et al., 2010) and Arabic WordNet (AWN) 2.0 (Black, et al., 2006). They evaluated these resources on a dataset of 2.3K documents.

ArSenL, (Badaro, et al., 2014) presented a publicly available large-scale Arabic sentiment lexicon (ArSenl) that consist of 29K lemmas with 158K synsets (group of synonyms). They mapped the Arabic WordNet (AWN) (Black, et al., 2006) with SentiWordNet and the mentioned SAMA with AWN. They assigned scores to AWN words through ESWN mapped synsets (synonym sets) and manually validated them. They normalized both SAMA and AWN to align their orthographies and mapped the words that have a minimum edit distance. Then they mapped SAMA's English words with ESWN and validated them by measuring the agreement with the first created lexicon and checking a random set of 400 lemmas. Finally, they took the union of the formed lexicons.

SLSA, (Eskander & Rambow, 2015) generated a publicly available sentiment lexicon for Standard Arabic containing 35K words. Copying the method of ArSenl they extracted polarity scores from SentiWordNet and mapped them with Arabic Morphological Analyzer (Aramorph) words by preprocessing the information provided from both lexicons which includes lemmas of Aramorphs' English glosses, normalized words based on their POS tags,

[11] https://en-maktoob.yahoo.com/ it seems that the Arabic version no longer existing.

and the average of the duplicate synset scores from SentiWordNet. They gave neutral scores to unmapped words. They evaluated the lexicon intrinsically and extrinsically achieving a slight improvement over the mentioned Arsenl.

(Mourad & Darwish, 2013) translated the MPQA sentiment lexicon (Wilson, et al., 2005) to Arabic. They used stemming, POS, and some Twitter tags as features. They evaluated their lexicon against a dataset of 2.3K tweets. (El-Makky, et al., 2014) took this lexicon and the lexicon of (Abdul-Mageed & Diab, 2011) to expand their Egyptian dialect lexicon.

(Al-Twairesh, et al., 2016) generated two lexicons automatically from a set of labelled Arabic tweets. For the first lexicon, they collected English word glosses that are equivalent to the words in the tweets, then cross checked it with two English lexicons: (Hu & Liu, 2004), and the MPQA (Wilson, et al., 2005). For the second lexicon, they searched for words that are semantically related to positive and negative tweets using PMI (Turney, 2002) measurement technique taking into account the frequency of the words as well.

3.2.2 Machine Learning Approaches

ML in Arabic sentiment analysis is a developing discipline, nevertheless we review the following works.

(Abbasi, et al., 2008) focused on extracting syntactic features such as vocabulary richness, word n-grams, and word roots from a labelled dataset of *in favor* or *against* a particular topic then they used an SVM classifier on two small datasets of 1K posts each. (Saleh, et al., 2011) tested SVM and Naïve Bayes (NB) classifiers using n-gram features on a labelled corpus of 500 movie reviews. Similarly, (Shoukry & Rafea, 2012) tested SVM and NB classifiers on a dataset of 1K tweets (500 positive and 500 negative) using n-gram features as well. (Itani, et al., 2012) proposed a Naïve Search (NS) using manually extracted features from text. They evaluated this approach on a labelled Arabic corpus of several dialects consisting of around 20K posts. (Al-Radaideh & Al-Qudah, 2017) tested SVM, K-NN (K-nearest neighbours), Decision Trees, and NB classifiers on the dataset of Shoukry 2013.

(Abdul-Mageed & Diab, 2011) generated ArabSenti a collection of 3.9K adjectives labelled

as positive, negative, or neutral. Then they extracted morphological and language independent features and fed them into different classifiers to evaluate their lexicon.

In (Salamah & Elkhlifi, 2014) three Kuwaiti natives annotated a large dataset of 340K political tweets. They extracted and compiled several linguistic resources and integrated them in several supervised classifiers. (Baly, et al., 2017) used words from different Arabic sentiment lexicons as features for SVM, logistic regression, and Random Forest Trees classifiers trained on a multi-dialect labelled dataset.

3.2.3 Deep Learning Approaches

Similarly, DL for Arabic sentiment analysis is developing as well.

(Dahou, et al., 2016) created a word embedding space from large Arabic corpus consisting of 3.4B words to train a convolutional neural network (CNN) model. They trained and tested their model on 5 different datasets: The LABR book reviews dataset (Aly & Atiya, 2013) which consists of over 63K reviews downloaded from Goodreads[12], Arabic Sentiment Tweets Dataset (ASTD) (Nabil, et al., 2015) which consists of over 10K Arabic tweets, Arabic Gold-Standard Twitter Sentiment Corpus (Refaee & Rieser, 2014) consisting of 2.3K tweets, another 2K tweets dataset (Abdulla, et al., 2013), and (ElSahar & El-Beltagy, 2015) that consists of 33K movie, hotels, and product reviews.

(Altowayan & Tao, 2016) created a word embedding space from a corpus of 190M words. They trained SVM and logistic regression classifiers with the obtained word representations as features on three Twitter labelled datasets consisting of 1.6K, 1.9K and 754 tweets.

(Al-Sallab, et al., 2017) trained a sentiment word embeddings using the mentioned lexicon ArSenl to assign sentiments to the vocabulary in the corpus. They fed the word representations to a Recursive Auto Encoder (RAE) model. They evaluated the model on three different datasets: 1.2K newswire sentences extracted from the Arabic Treebank (ATB) (Maamouri, et al., 2004), 1.1K online comments extracted from the Qatar Arabic Language

[12] https://www.goodreads.com/

Bank (QALB) corpus (Mohit, et al., 2014)and 2.3K tweets dataset (Refaee & Rieser, 2014). They achieved better sentiment analysis results over using the RAE model without the sentiment embeddings.

(Baly, et al., 2017) introduced new features for the Recurrent Neural Tensor Network (RNTN) by (Socher, et al., 2013) for Arabic sentiment analysis. They built an Arabic sentiment tree bank that is enriched with different combinations of morphological abstractions of words and orthographic representations and used it in the RNTN model. They also created an annotated dataset of around 1.2K comments (ArSenTB). They trained their model on the dataset to achieve an improved score over the RNTN with a basic tree bank.

(Al-Azani & El-Alfy, 2017) trained word embeddings from around 190M words. Then, they tested four LSTM RNN models trained on a Twitter dataset of 1.8K tweets.

- A simple LSTM
- CNN-LSTM: A CNN layer added to the LSTM
- Stacked LSTM: Three LSTM layers stacked on top of each other
- Combined LSTM: A combination of two LSTMs.

The combined LSTM achieved the highest score in sentiment classification.

(Farha & Magdy, 2019) developed a DL model that feeds word embedding information into a neural network of CNN and LSTM. They created the word embeddings from a large corpus of 250M tweets. They evaluated the model on three different datasets consisting of around 10K, 17K, and 18K tweets achieving a small improvement over the state of the art.

3.2.4 Discussion

We follow by discussing each of the mentioned lexicon based, machine learning, and deep learning approaches for Arabic sentiment analysis separately in the following subsections.

3.2.4.1 Lexicon Based Approaches

As can be seen from the literature of Arabic lexicon-based sentiment analysis, most efforts

are focused on MSA. Arabizi however is a transcription of DA, a different variety of Arabic. Before building a new sentiment lexicon for Arabizi, we tried to exploit the SANA sentiment lexicon (Abdul-Mageed & Diab, 2014), which is to the best of our knowledge the only published sentiment lexicon that consists of Levantine dialect among other dialects as claimed in the paper. Unfortunately, the developers of this resource did not publicise it or share it with us. On the other hand, Ramitechs[13], a lexical resources company owned by Ramy Eskader, the author of the mentioned SLSA lexicon (Eskander & Rambow, 2015) offered us a Levantine lexicon for an infeasible price. As such, we built a new Lebanese dialect Arabizi sentiment lexicon over two phases, generation and expansion (Chapter 5).

Similar to the related works, in the first phase we deployed some translation and manual selection steps to create a list of Lebanese Arabizi sentiment words. In the second phase however, unlike the reviewed lexicons, we enriched the generated sentiment words with their word forms using the word embeddings deep learning technique to address the sparsity challenge of Arabizi (Chapter 2).

3.2.4.2 Machine Learning Approaches

As mentioned earlier, ML approaches are data-driven; with the current lack of Arabizi annotated data and the high cost of creating such data (Chapter 2), satisfying the conditions of ML approaches for Arabizi becomes very expensive. The inconsistent orthography of Arabizi makes the language highly sparse (Chapter 2), such that the size of the training data that sufficed to train a ML approach for Arabic might not suffice for Arabizi.

In the latter works in ML, they combined lexicons with ML approaches. (Baly, et al., 2019) for example, used the words in a sentiment lexicon as features to train a ML approach. As if the ML approach is being informed about the important words for polarity classes.

First, this presents a new evaluation technique of the sentiment lexicon. The lexicon may be evaluated based on whether the ML classifier improves the classification with the lexicon words as features. Second, this highlights some of the benefits of a new lexicon outside the

[13] http://www.ramitechs.com

scope of lexicon-based approach.

3.2.4.3 Deep Learning Approaches

As can be seen, these data-hungry deep learning neural network architectures have been trending lately in NLP for their powerful performance. The concept is to prepare a NN model initially by training it on a large amount of unsupervised texts and fine tune it later for sentiment analysis. These works have trained different models with different parameters on different datasets to finally test them for sentiment analysis. They evaluated these models against annotated datasets for Arabic ranging from 754 tweets in (Altowayan & Tao, 2016) to 18K tweets in (Farha & Magdy, 2019).

Although in this thesis we plan to create a dataset for evaluation, which gives us the opportunity to try similar DL approaches, these models have been trained initially on large amount of text such as the 190M words (Altowayan & Tao, 2016) and 3.4B words (Dahou, et al., 2016). Collecting an Arabizi dataset of such sizes from social media could be very costly, although unsupervised, Arabizi is mixed with English in Lebanon Twitter data at a small ratio of 1:7 (Chapter 2). One of the strengths in the pipeline of our work is the development of an Arabizi identification approach to automatically select Arabizi sentences from English which may be used to create datasets as large as the ones in the mentioned works to train NN models as a future extension of this work (Chapter 4).

3.3 Arabizi in NLP

In this section we review some of the literature on NLP for Arabizi in detail. We start by reviewing efforts on automatic transliteration then focus on little works that did sentiment analysis for the transliterated Arabizi.

3.3.1 Transliteration

Transliteration as mentioned in Chapter 2 is the automatic conversion of words written in Latin script, Arabizi, into the Arabic script, Arabic. Since Arabizi is considered one form of written dialectal Arabic, several researchers saw that it could be converted into the Arabic script. However, this task is not a straight-forward character replacement because there is no unified orthography for Arabizi (Chapter 2).

(Masmoudi, et al., 2015) focused on Tunisian dialect Arabizi. They proposed a handcrafted rule based transliterator that generates several transliterations for every Arabizi word. They then normalised the text and manually selected what they thought was a correct transliteration. Finally, they evaluated the transliterations by calculating the percentage of agreement between the users' choices and the transliterator's output. We did not find the evaluation very clear because the transliterator generates several transliterations, however they presented a good error analysis of character ambiguities.

(Chalabi & Gerges, 2012) proposed another rule based transliterator that generates several transliterations for each word as well. They scored and ranked the candidate transliterations using word and character language models. They then introduced a stemming phase without explaining or referring to the process. They claimed that they added all possible affixes to the words, although Arabic is rich in morphology where the root could be altered as explained in Chapter 2. The dialect they chose is unknown and it is not clear which dataset they used for evaluation. They claimed a 90% accuracy without demonstrating examples or errors.

(Darwish, 2014) created a manually transliterated Egyptian Arabizi-Arabic corpus composed of around 3.4K words extracted from Twitter. They did some light normalization on the text, aligned the word pairs using GIZA++[14], and generated a list of candidate transliterations for each input word. They tested their model on 1.3K words. They mapped the candidates with a large corpus of 112M tweets to select the candidates that appeared the most achieving a transliteration accuracy of 88.7%. They presented a clear error analysis with examples.

(Al-Badrashiny, et al., 2014) built a highly sophisticated system to achieve 69% transliteration accuracy for Egyptian Arabizi, named *3Arrib*[15]. They passed the input text

[14] http://www.statmt.org/moses/giza/GIZA++.html
[15] https://camel.abudhabi.nyu.edu/arrib/index.html transliterator is not working (Nov. 2019).

through several preprocessing steps then fetched it into a finite state transducer (FST) that also generates a list of possible transliterations. The FST is trained using around 8.5K pairs of words aligned on GIZA++ as well. The existence of words is cross checked with a pre-defined system called CALIMA (Habash, et al., 2012). They then applied another series of complicated text preprocessing and tokenization to the text. Similar to (Darwish, 2014), they built a language model but from 392M words, that were also preprocessed, to search for and select the most frequent resulting transliteration. They tested this system on 1K words and presented an error analysis with examples.

(May, et al., 2014) did a similar work in concept to that of (Al-Badrashiny, et al., 2014) also for Egyptian Arabizi however using two weighted finite state transducers (wFST) and way less preprocessing and normalisations. They collected a corpus of around 800 hand-aligned word pairs. They aligned the characters in 3K sentences using GIZA++ as well. The weights obtained from the first wFST represents the conditional probability of the given character. They maximized the probability for the Arabic output then added word pairs reduced in length, without the vowel letters for Arabizi. They finally used a BLEU scoring method (Papineni, et al., 2002) to measure the similarity with a referred translation for an intrinsic and extrinsic evaluations.

(Guellil, et al., 2017) proposed the first neural networks approach for transliterating Arabizi of Algerian dialect. To facilitate the process of creating a corpus of parallel text, they started by creating a rule-based transliterator to transliterate 1.3K sentences that were manually fixed afterwards. They trained a neural machine transliteration (NMTR) model on the corpus and on a lexicon that is merely defined. They aligned a lexicon of weighted characters using the method of (Neubig, 2016). They also used an LSTM layer to train the model. They finally tested the results of the trained NMTR using several epochs achieving accuracies of 45% and 73% on external and internal datasets respectively of size 1K sentences. They presented a good error analysis with examples.

3.3.2 Sentiment Analysis

(Al-Aziz, et al., 2011) claimed that the differences in Egyptian Arabizi orthography can be unified if we encode Arabizi with numerals. First, they preprocessed Arabizi with heavy

normalisations. They then created a table of equivalent Arabic and Arabizi characters. They set the characters into groups and assigned each group a number. To prove their claim, they asked five Egyptian natives to transcribe Arabic script sentiment words (170 positive and 582 negative) into Arabizi. By that they would have several orthographies for each word allowing them to measure the coding similarity among the resulting Arabizi transcriptions claiming that words of different orthographies that share the same code is useful for sorting out the inconsistent orthography issue of Arabizi. In fact, this worsens it by increasing the number of possible words every code could generate. For example: *farem* - *chopping* and *barem* - *spinning*, *7abib* - *beloved* and *3afif* - *dignity* would result in the same code though they differ in meaning and spelling. Although asking five natives to transcribe sentiment words in Arabizi is a nice way to encompass some differences in orthographies, it requires manual effort to result in a small number of differences. In addition, the generated list was not tested for sentiment analysis nor made public.

(Mataoui, et al., 2016) developed a lexicon-based approach for sentiment analysis of the Algerian Arabic social media text while taking Arabizi into account. They collected a FB corpus of 7.6K comments from 200 posts. The comments in the corpus are distributed as follows: 1.5K MSA, 2.4K Algerian Arabic, 1.9K Arabizi, and 1.2K foreign languages mostly French. They built a sentiment lexicon by manually converting MSA and Egyptian dialect sentiment words[16] to Algerian dialect, a total of 713 positive and 2.3K negative words. They preprocessed and normalized the text. They used Google translate to translate any French and transliterate Arabizi words detected in the text. They used a simple word scoring method alongside many handcrafted rules. We did not find it clear how much of the corpus they annotated and how did they annotate it, but assuming they annotated the 7.6K comments and tested their approach against it, they presented detailed results of their approach. They achieved a baseline accuracy of 53%, 65% with Arabizi transliteration, 72% with French translation, and 79% by adding sentiment phrases to the lexicon and stemming[17] Arabic. This is summarized in the Table 3.1. The recall, precision, and F-scores were not presented.

[16] Unknown sentiment lexicons produced by Nile University.

[17] We note that public stemmers for Arabic are known for their naïve stemming and blind reduction of affixes that could be root letters such as the one used in this work, Khoja Stemmer (Khoja 1999) http://zeus.cs.pacificu.edu/shereen/research.htm#stemming. It is also designed for MSA, the authors in this work did not explain how did it perform on Algerian dialect Arabic.

	Baseline	With Arabizi transliteration	With French translation	With stemming and adding phrases
Accuracy	0.53	0.65	0.72	0.79

Table 3.1: Impact of adding Arabizi to Arabic sentiment analysis

A major drawback of this work is not evaluating the transliterations of Google translate. In Chapter 2 we pointed out to the expected high percentage of erroneous transliterations due to the word ambiguity challenge due to the inconsistent orthography. On a side note, it is inaccurate to assume from their data that 26% of the social media text in Algeria is Arabizi since the collected FB corpus is biased to selected pages of certain genre, however, the accuracy of the lexicon-based approach improved by 12% for transliterating Arabizi given it consists of 26% of the data. This proves that analysing Arabizi leverages the sentiment analysis for Arabic.

(Duwairi, et al., 2016) present a limited work on sentiment analysis for Jordanian dialect Arabizi transliterations. They collected 3.2K Arabizi tweets and manually annotated them for sentiment. They created a rule-based transliterator that maps every Arabizi character with an Arabic script character. They did not present the mapping table of their transliterator and did not evaluate the resulting transliterations. The complexity of transliterating Arabizi is clearly shown in the transliteration works mentioned in the previous section e.g. (Al-Badrashiny, et al., 2014). They then claimed that they applied NB and SVM algorithms to classify tweets into sentiment classes without providing any details on the training, testing, and feature selection. They also claimed that they applied subjectivity classification without any details as well. They finally displayed recall and precision results for positive and negative classes separately. Results for the positive class were unknowingly significantly higher than those of the negative class which are below 50%. No discussion or examples of classification and errors were presented.

Finally, (GUELLIL, et al., 2018) proposed an interesting pipeline to classify Algerian Arabizi data into sentiment classes. They translated SOCAL (Taboada, et al., 2011), an English adjectives lexicon of 2.8K words with polarity scores ranging from very negative -5 to very

positive +5, to Algerian dialect Arabic automatically using Glosbe API[18], which to their luck contains Algerian Arabic dialect. For every English word in SOCAL they took the translated word or set of words (synsets), and gave them the same score as that of the English word. They reviewed the resulting lexicon manually to have 771 positive and 968 negative words. This lexicon was not used for classification, but for annotating a corpus automatically to train word vectors and ML algorithms. They annotated a FB corpus balanced with 127K positive and 127K negative messages. It is not quiet clear but it seems that they took part of the automatically annotated data to evaluate an Algerian Arabizi transliterator that they created as well. The transliterator generates all possible orthographic Arabic variances for every Arabizi word using handcrafted set of letter to letter mappings, then a language model chooses the best candidate based on its frequency of occurrence from a large corpus similar to the mentioned works in the the transliteration section above (Masmoudi, et al., 2015), (Chalabi & Gerges, 2012), (Darwish, 2014), (Al-Badrashiny, et al., 2014). They transliterated a dataset manually to evaluate the performance of the transliterator claiming an accuracy of 72% without presenting evaluation details. Next, they used both datasets, the automatically and manually transliterated, to create an embedding vector space and train ML classifiers. They fetched the vectors into the classifiers as input features. They experimented this with several classifiers achieving the highest F1-scores of 76% and 75% using NB and Random Forests Trees on the automatically transliterated dataset and 78% and 77% respectively on the manually transliterated dataset. They finally presented some error analysis.

3.3.3 Discussion

Although Arabizi is seen by most researchers in the reviewed work as a form of Arabic that should in one way or another be converted to Arabic script, its natural inconsistent Latin script introduces several linguistic complexities that might be very difficult to address heuristically in normalising text and handcrafting rules to satisfy minor observations in Arabic. Constantly changing the natural language produced by us using basic rules, writing conventions, and normalisations in an attempt to simplify the language might give shallow solutions but risks causing deeper complexities that are beyond our current perspective. For example:

[18] https://glosbe.com/a-api

The article in Arabic is the attachment of the proclitic ال *al* to the word:

the + pen: the pen alkalam القلم: قلم + ال pen *kalam* :قلم

Hand crafting rules to strip the articles from words, one might blindly go down in this spiral:

1. *Remove* the article ال *al from beginning of word*: Then the tri-literal word الم *alam* comes up which means *pain*, where the beginning ال *al* are root letters.

2. *If word length is 3 or less, skip it, otherwise strip* ال *al from beginning of word:* Then the quadri-literal word اليف *aleef* comes up which means *harmless* or *domestic*, a positive adjective for pets, also ال *al* are root letters.

3. *if word length is 4 or less, skip it, otherwise strip* ال *al from beginning of word:* Then the tri-literal word عزّ *3izz - glory* whether written with or without the *shaddah* (gemination) but with an article العز would bypass the rule.

Leading to an endless loop of rules and exceptions.

Same applies for morphology, the following three words are all in the plural form, they all share the same pluralization pattern, but the singular form for each has a different pattern.

قضاة *judges*	جهات *directions*	مواد *materials*
قاضي *judge*	جهة *direction*	مادة *material*

There are also words in singular form that has the same pattern of the mentioned plural forms such as صلاة *prayer*.

The word تقعون *you fall off (2nd person) plural* derives from the root word وقع where the و *waw*, a root letter, is dropped in the inflection.

As such, the morphology and orthography in Arabic is beyond miniscule normalisations. Majority of the works in the literature on Arabizi focus on transliterating it to Arabic thus going through preprocessing that includes heavy normalisation catered to one dialect. Then

mapping the Arabizi characters with Arabic letters by following a mapping table that was set heuristically as well.

First, setting up the evaluation datasets and parallel corpora requires a major effort in manual transliteration yet there is no unified orthography for Arabizi, hence a model trained or evaluated on a dataset written by an individual is trained exclusively to the orthographic style of this individual.

Some of these works addressed the challenge of inconsistent orthography by aligning characters and training FSTs to predict a list of transliteration that has to go through a language model to select the best candidate based on the frequency of the predicted words in large corpora, although the more frequent words are not necessarily the correct ones.

Second, preprocessing the text extensively and handcrafting rules for a specific dialect from an individualistic perspective might degrade the value of the NLP research. Most efforts consider Egyptian Arabizi solely; as can be seen in Chapter 2, the ambiguity of Lebanese Arabizi is higher than that of the Egyptian having less consonant letter representations. As such preprocessing efforts for one Arabizi dialect might not fit for other dialects. However, Egyptian is the most spoken Arabic dialect, hence the value of these works is apparent.

Finally, far from the complexities of transliterating Arabizi into Arabic script, if the target dialect is as under-resourced for sentiment analysis as Arabizi then transliteration efforts might not add value for sentiment analysis. However, it could be used to unify the written natural Arabic language to a single script. Hence future advancements for written dialectal Arabic would cover Arabizi.

In this thesis we take a total different direction. Instead of diving into complex transliteration attempts, we perceive Arabizi as if it is an under-resourced language independent of Arabic. We aim to create resources to make it possible to analyse the sentiment from this text, directly, without the need to formulate mapping rules to transliterate it to Arabic or to change its natural form by any means.

We address the linguistic issues of rich morphology and inconsistent orthography that relies immensely on NLP resources using word embedding to automatically find naturally written

orthographic and inflectional variants from a collection of Arabizi data posted on the social media. We explore different ways to use the word embeddings to maximise the coverage of the sentiment lexicon without heavy preprocessing of the raw data. Instead of catering linguistic rules for an Egyptian or Jordanian or Algerian dialect, one of the advantages of dealing with Arabizi directly is the possibilty to reproduce the work applied on one dialect onto other dialects.

3.4 Discussion

Unlike the works of sentiment analysis for standard languages we are dealing with an extremely low-resourced, highly sparse texting language that is prominent on Arab social media. For that, as promising the recent neural network architectures are, the inevitable fact is that they are data hungry, driven by large amounts of training data, an essential requirement that is simply infeasible in low-resourced languages. For that, the direction we take in this research focuses on building new resources for the sentiment analysis of Arabizi. Since the nature of the language is highly sparse, its vocabulary is very large, we utilize the neural network architecture (word embeddings) for resourcing Arabizi not for sentiment classification. After resourcing a new sentiment lexicon and creating sentiment annotated dataset, we evaluate the newly created resource using a classical sentiment analysis approach, lexicon-based. We acknowledge that this approach is basic but valuable in the current context of no-resources.

On another front, several researchers saw that Arabizi has to be transliterated to Arabic in a way or another hence focused their efforts around this task. We found throughout our study of Arabizi that its linguistic complexities are beyond any straight forward automatic de-Latinisation approach reported earlier (Chapter 2). We also learned that most research on Arabic sentiment analysis targets MSA, and recently DA mainly Egyptian and North African but not Lebanese to the best of our knowledge. We anticipated that transliterating Arabizi accurately is a difficult task, yet if successful, would lead us to another low-resourced language domain. Therefore we decided to resource Arabizi as a new language independent of Arabic. Given that Arabizi consists of several linguistic challenges that are common to other languages such as inconsistent orthography and rich morphology, we aspire that the

value of our work would be reflected in future efforts on resourcing other low-resourced languages.

Throughout this work we created several datasets and trained a language identifier but the core of the thesis lies in the development of a new morphologically and orthographically rich sentiment lexicon (Chapters 5 and 6).

Most of the mentioned lexicons in Section 3.2.1 for Arabic sentiment analysis such as Sifaat (Abdul-Mageed & Diab, 2012), SANA (Abdul-Mageed & Diab, 2014), ASWN (Alhazmi, Black, & McNaught, 2013), ArSenL, (Badaro, et al., 2014) and SLSA, (Eskander & Rambow, 2015) are comprehensive types of lexicons that exhaust a large number of words with sentiment scores. These types of lexicons include positive, negative, and neutral words. Some translated English sentiment words or extracted them from Arabic sentiment labelled data, while others extended existing Arabic lexicons by mapping them with Senti and Arabic WordNets (Esuli & Sebastiani, 2007), (Black, et al., 2006) to compute sentiment scores.

Since Arabizi is low-resourced our goal was to find words that are exclusively positive and negative as a first step towards building the new sentiment lexicon, for that we also utilised the translation technique from English but from different resources. We used the (Hu & Liu, 2004), MPQA (Wilson, et al., 2005), and another private dialectal Arabic word list followed by phases of manual selection detailed in Chapter 5. The motive for this is to minimise the manual effort in selecting which words are dialectal and which are positive or negative, thus going through lists of few thousand translated words rather than 28.7K MSA words such as ArSenL (Badaro, et al., 2014).

However, we know that most Arabizi sentiment words could be inflected in a wide range of forms, of which each could be transcribed in various ways, as such after generating a new list of positive and negative Arabizi words, unlike the mentioned literature, we focus on addressing the lexical sparsity. We explore word embeddings to expand the generated sentiment words to their wider range of forms.

Sentiment analysis for Arabizi is still at its infancy, we therefore aspire that our contributions motivate the Arabic NLP community to build upon our work for Arabizi since it constitutes 6% of Twitter's data in some regions and is proven to be common among the youth (Chapter 1).

3.5 Chapter Summary

In this chapter we surveyed the literature of sentiment analysis. We started by highlighting and explaining some popular approaches used to analyse sentiment such as deep learning and word embeddings in general. Then we narrowed down to review and discuss the literature of sentiment analysis for Arabic giving attention on the lexicon based approaches as they relate to our research. We finally focused on what other researchers have done to process and analyse Arabizi. We presented the drawbacks of handcrafting rules to process and transliterate Arabizi and some of the advantages of treating it as a language independent of Arabic.

II. Resources

4 Data Collection

طرقت الباب حتى كلّ متنى فلما كلّ متنى كلمتنى

فقالت لى يا اسماعيل صبرا فقلت يا أسماعيل صبرى

Ismail Sabri Pasha

Despite the usage of different dialects of Arabizi on social media and mobile messaging (Chapter 3), to the best of our knowledge there are no publicly available Arabizi data resources for NLP tasks such as large parallel corpora for transliteration, sentiment-annotated data for sentiment analysis, or a tree bank for parsing. The lack of such public resources marks this written language as a low or under-resourced language.

This chapter presents the creation of two annotated datasets and a corpus. We refer to an annotated dataset as a collection of social media text that has been annotated by humans. We use the first dataset to train an Arabizi identifier and the second to evaluate the sentiment analysis approach proposed in this thesis. The proposed Arabizi identifier would help us in harvesting a large corpus of Arabizi conversations. A corpus in general is a large compilation of written texts covering a particular subject. In this context, we refer to the corpus as a compilation of Arabizi text. We will use the corpus to discover inflectional and different orthographic variants of the sentiment words found in the lexicon proposed in Chapter 6. We detail the annotated datasets in Section 4.2 and the corpus in Section 4.3.

Resourcing Lebanese dialect Arabizi with datasets and a corpus not only contributes to this research, but also to other NLP tasks such as training language models, creating tree banks and parts of speech (POS) parsers.

4.1 Introduction

(Bies, et al., 2014) mentioned that the use of Arabizi is prevalent enough to pose a challenge for Arabic NLP research. She also mentioned that there are no naturally occurring parallel texts of Arabizi and Arabic script. In (Bies, et al., 2014) they developed a parallel Egyptian dialect Arabizi-Arabic corpus of 3.2K SMS messages by manual transliteration. We note that this developed corpus is not public, it is not annotated for sentiment, and it is Egyptian dialect Arabizi. It is difficult for this resource to satisfy our need for sentiment analysis and it is quiet different in dialect from the case dialect that we study in this thesis; Lebanese Arabic. As mentioned in Chapter 2, the written Lebanese Arabizi differs from the Egyptian in orthographic style, choice of letters, and a major choice of words which makes the dialectal differences.

Given the lack of Arabizi resources for NLP in general, and Lebanese in specific, collecting Arabizi data had become necessary for the course of this research. In this thesis, we propose to analyse sentiment directly from Arabizi text. We focus on the creation and expansion of a sentiment lexicon to achieve this goal. Therefore, Arabizi data is integral for these steps: creating an Arabizi corpus, expanding the lexicon, and evaluating the performance of the lexicon.

The method of the evaluation is a comparison of the output of the proposed approach against a human decision or assignment. If the output class matches with the sentiment class assigned by a human for a given text, then the output of the approach is considered a success in this case, and a failure otherwise. For example:

If the sentiment analysis approach classifies the following sentence as negative or neutral, but the human annotators agreed that it is positive, then the approach fails in classifying the sentiment of this tweet: *Guys ana nezil 7areb w 2oul la2 lal fased / Guys I am going to fight and say no to corruption.* Doing this across all sentences in a human annotated dataset gives us an idea of how well the proposed approach is performing.

On the other hand, a large collection of Arabizi conversations is also required to explore the different morphological and orthographic variations that are used by the users. This is needed to maximise the coverage of sentiment words. As mentioned in Chapter 2, since Arabizi lacks

a consistent orthography, one sentiment word could be written in several ways. For example, *khayr / good*: *kheir, kher, khyr, khair, kheyr, khyr, 5eir, 5yr, 5ayr, 5er*… Hence a large corpus of natural Arabizi conversations could help us discover the orthographic variants of sentiment words. The lack of orthographic consistency gives Arabizi a high degree of lexical sparsity, a challenge for matching social text with sentiment lexicon, thus maximising the number of orthographic variants per word decreases the degree of lexical sparsity that would potentially improve sentiment analysis.

However, since Arabizi is expressed by bilinguals from Arab countries (Chapter 3), it is usually found within multilingual messages and it is codeswitched with Latin script languages as well, as shown in the pilot study in Chapter 2. The codeswitching in Arabizi is therefore inter-sentential and intra-sentential.

Inter-sentential: Codeswitching is bounded by the sentences, one sentence could be written in English and the other in Arabizi. This could happen from several users in a conversation or from a single user. For example:

User 1: *how is it going?*
User 2: *tamem, w enta? / fine, and you?*

User: Welcome back, nawwar lebnen / welcome back, Lebanon just got brightened.

Intra-sentential: Codeswitching occurs middle of sentence with no interruptions or separations such as a comma or period to indicate a codeswitch. For example:

keep a 3aj2a kit ma3ak matra7 el first aid kit in case 3le2et bi 3aj2a / keep a traffic kit with you with the first aid kit in case you got stuck in traffic.

As such harvesting Arabizi data requires isolating Arabizi from other languages, known as language identification. Arabizi identification could be applied automatically by a classifier trained on annotated text Arabizi/Not-Arabizi.

4.2 Annotated Datasets

We will address RQ2 in Chapter 6: *How can an Arabizi lexicon be developed and used for sentiment analysis.* We propose a new Arabizi sentiment lexicon and evaluate the sentiment analysis performance of this lexicon using a lexicon-based approach. However, a prerequisite to this evaluation is creating a sentiment-annotated dataset.

We use Twitter as the source of the data to create the annotated datasets. Twitter is an online social network that is quiet active in Lebanon based on the pilot study done in Chapter 2. Twitter users express themselves in short texts limited to 280 characters, tweets, to be shared and interacted with publicly.

In Chapter 3 we showed that the usage of Arabizi is more frequent in private mobile messaging than on social media, however, the pilot study in Chapter 2 showed that 53% of Lebanon's tweets in 2016 are Latin script of which 9.3% are Arabizi. Hence another reason for choosing twitter, it is a public platform and contains Arabizi data. Twitter data collection is a simple task via the API[19] provided by Twitter.

In this section we create two Twitter datasets:

1. Arabizi identification (AI) dataset.
2. Sentiment analysis (SA) dataset.

The first dataset consists of tweets labelled as Arabizi or Not Arabizi; we will use it in Section 4.3 to train a Language Identifier to identify Arabizi from other Latin script languages. The second dataset consists of Arabizi tweets labelled as positive, negative, or neutral; we will use it in Chapter 7 to evaluate the sentiment analysis lexicon-based approach. The creation of these datasets is described in four subsections: data collection, preprocessing, annotation, and results.

[19] https://developer.twitter.com/en/docs/tutorials/consuming-streaming-data

4.2.1 Data Collection

One approach of collecting Tweets from twitter is using the API to search a keyword and retrieve tweets that contain this keyword. Since we plan to collect Lebanese dialect Arabizi for this research we opt out from using this approach to minimise the risk of retrieving Arabizi tweets of other dialects, also, not to limit the data to tweets containing certain keywords. Instead, we used Twitter stream API to collect live tweets, as they get tweeted within a specified region, Lebanon in this case. Similar to the Twitter data collection in the pilot study in Section 2.1, we specified the API with geographic coordinates to cover the region of Lebanon. The API takes two geo-coordinates and streams all tweets coming from within the specified strip. We used (33.5, 33.36) and (34.22, 35.96) to cover all the region surrounding Beirut, the capital city of Lebanon.

We ran the Twitter stream script for a total period of around 4 months, March, and July to September of 2016 collecting a total of 177K tweets.

Twitter API provides meta information with each collected tweet such as the tweets' and users' ID numbers, tweet location and language, hashtags, mentions, user language, and number of followers and users following. In Figure 4.1 we present a snapshot of three tweets of different languages from Lebanon: Arabizi, English, and Arabic.

7083613551...	@ChaibanRaja raja btotlaale wen maken...	None	ChaibanRaja	ht	لبنان	LB	Lebanon	0	YasminaBallout	44462...	635	378 en
7083614155...	math is getting complicated	None	None	en	لبنان	LB	Lebanon	0	fanboyology	324320...	86	406 en
7083614253...	حكم مباراة تونسي بيكي بسبب شتم الجمهور لوالدته	None	None	ar	لبنان	LB	لبنان	0	kalimahorra1	281119...	2140	2254 ar

Figure 4.1: Tweet with Meta Info

Twitter API was able to detect Arabic but not Arabizi tweets. It misidentified Arabizi for (ht, tr, in, hi, pt, nl, ct, or ey) languages, where some are known and stand for (Haitian, Turkish, Hindi, Portuguese, and Dutch). Using these unrelated language tags by Twitter API to identify the Arabizi tweets was insufficient because the API also misidentified many tweets that contained URLs and informal expressions such as *lool, hahaha,* or repeated letters within words. For example:

Arabizi Tweet: *@abdlsater ahla w sahlaaaa?* API Lang: tr

English Tweet: *#morning #selfie #smile #beard #blond* API Lang: tr
#blue #home #kaslik #lebanon
@Kaslik https://t.co/g7j5rUvgG7

We filter out the Arabic tweets that were identified by Twitter API as Arabic, around 80K. The remaining dataset contains 97K Latin script tweets. To accurately identify the Arabizi ones, we preprocessed the dataset and resorted to a manual annotation task.

4.2.2 Preprocessing

Twitter granted the public a social space to express themselves in a short text limited to 280 characters. While our interest lies in tweets composed of words, to analyse sentiment, many tweets are composed of symbols, URLs, images, or videos. In this step, we attempt to maximise the Twitter data that contains an alphabet, which indicate that the data is composed of words, by filtering out tweets that lack an alphabet automatically. We also clean the Twitter data that contain alphabet by removing URLs, hashtags (words preceded by #), mentions (words preceded by @), and non-ASC characters.

Usually hashtags and mentions in tweets are used to indicate the theme, location, time, or persons. For example:

Tweet: *wish could go back in time to the good old days #buenosaires #2007*

As such, hashtags and mentions could serve as valuable features for named entity recognition (NER) or sentiment analysis with targets to identify the entity that the sentiment is targeted upon. However, this is beyond the scope of this research at this stage. In this thesis we focus our efforts on analysing sentiment from Arabizi data as a first step towards direct analysis of Arabizi without prior transliteration attempts. Therefore, the desired output after analysing the mentioned tweet is *positive*, regardless of the hashtags. For example:

Tweet: *wish could go back in time to the good old days #buenosaires #2007*

Desired output: positive

Tweet: *wish could go back in time to the good old days #zanzibar #2017*
Desired output: positive

Preprocessed: *wish could go back in time to the good old days*

Filtering tweets from hashtags, mentions, URLs, and non-ASC characters resulted in many tweets lacking an alphabet, these are tweets that were not composed of words originally. For example:

Tweet: @najwakaram ? https://t.co/WkS2XnHhji
Preprocessed: ?

We removed all such tweets from the data. We also kept one copy of tweets that are duplicated (twice or more). For example, one of the following preprocessed tweets would remain in the data.

Tweet: *wish could go back in time to the good old days #buenosaires #2007*
Preprocessed: *wish could go back in time to the good old days*

Tweet: *wish could go back in time to the good old days #zanzibar #2017*
Preprocessed: *wish could go back in time to the good old days*

Finally, proprocessing reduced the collected Lebanon Latin script Twitter data from 97K to 66K tweets.

4.2.3 Annotation

In this section we describe the creation of the annotated datasets in six subsections: annotators, dataset, setup, platform, instructions, and results.

4.2.3.1 Annotators

We needed at least three Lebanese natives for the annotation task. Three annotators help break the tie, if two annotators disagreed upon a tweet, meaning if each one annotated a tweet differently, the third annotator breaks the tie. Thus, giving us the option to create datasets of two or three annotator agreement. For example:

What is the sentiment of the following tweets?

Tweet1: *bheb keef fe aalam ma baarefa btaaref kteer eshya 3anne*
I like how there are people whom I don't know know a lot about me

Annotator 1: positive Annotator 2: negative Annotator 3: positive

Tweet2: *saba7 l kheirrr ya habibit albiii nchalla ykoon nharek 7ilo*
good morninggg darling hope you have a nice day

Annotator 1: positive Annotator 2: positive Annotator 3: positive

Tweet 1 has a two annotator agreement and tweet 2 has three annotator agreement for the sentiment class positive.

Since the Twitter data is in Lebanese, we preferred the annotators to be Lebanese natives to relate to the esoteric dialectal expressions. Three undergraduate Lebanese students volunteered 30 hours for the annotation task. We trusted these students for this task based on their academic performance and the recommendation received from their supervisor.

4.2.3.2 Dataset

To abide by the annotation volunteering time, we had to limit the number of tweets for the annotation task. We conducted a test annotation of 1K tweets to observe the annotation quality (Section 4.2.3.5) and estimated the time it would take to annotate a larger set of twitter data. 1K tweets took around 60 minutes to annotate, that is 3.6 seconds per tweet on

average. Out of the collected and preprocessed 66K Latin script tweets from Lebanon (Section 4.2.1), we part 30K tweets randomly to fit in the annotation timeframe of 30 hours.

Based on the observation from the pilot study done in Section 2.1, 9.3% of the Latin script tweets are Arabizi from a sample of 5K Latin script tweets streamed from Lebanon, reported in Section 2.1.2. It is therefore expected to obtain a new dataset of around 3K Arabizi tweets from 30K tweets. A relatively small dataset that we use to evaluate the lexicon based approach and benchmark the results for further analysis.

4.2.3.3 Setup

To create the two datasets; AI and SA, we first need to know whether a given tweet is Arabizi. If the tweet is Arabizi, only then we would need to know what is the sentiment of that tweet. Therefore, the sentiment annotation depends on the script of the tweet (whether it is Arabizi or not). Hence, both datasets are interconnected with each other and could be created in a single annotation task. We set two annotation questions for each tweet:

1. Is the tweet written mostly in Arabizi?
2. What is the sentiment of the tweet?

All annotators have to annotate all tweets by answering these questions. The result of the first annotation question should produce the AI dataset for Arabizi identification and the result of the second annotation question should produce the SA dataset for sentiment analysis evaluation.

4.2.3.4 Platform

We created an annotation platform to assign the annotation task to the students. Although crowdflower[20], a public paid service that connects annotation tasks with annotators, was available during the time of the annotation, there was a major limitation. Back then, end of 2016, crowdflower offered users to design tasks and set annotators criteria, the service would

[20] Now known as figure eight https://www.figure-eight.com/

crowdsource random annotators that are registered with crowdflower to match the criteria. Assigning the annotation task to specific annotators was not an option, also choosing annotators from Lebanon was not an option either. Hence, we created an annotation platform mainly to design the annotation task and assign it to our recommended volunteers. Additionally, saving the annotation cost of public services.

Another known annotation service, Mechanical Turk[21], was not running during the time of the task.

Although at this stage we were looking at identifying the Arabizi tweets from the collected and preprocessed Latinscript Twitter data, sentiment analysis of Arabizi data is the main objective that drives this thesis. Therefore, after identifying the Arabizi tweets, they need to be annotated for sentiment as well. One annotation task was designed for both purposes: to identify Arabizi tweets among other Latin script languages, and to label these tweets with sentiment labels (positive, negative, or neutral).

We designed a simple annotation platform that displays the tweets in random order for each annotator. It asks the annotator *Is the tweet written mostly in Arabizi?* If the annotator answers *yes* for Arabizi, it then asks them *what is the sentiment of the tweet?*

As explained in Chapter 2, Arabizi is codeswitched, where users alternate with Latin script languages as they text.

The meaning of *mostly Arabizi* in this case, if codeswitching occurs in a tweet, the language of the tweet would be the dominating language which clearly comprises the majority of the words. For example:

Tweet: *Please ma ba2 thotto di3ayit Amir El Layl ugh*
Majority of words are Arabizi with one English word. This is considered an Arabizi tweet.

Tweet: Mafi master's in bioinfo. Not in LAU at least None
Majority of words are English with one Arabizi word. This is considered an English tweet.

[21] https://www.mturk.com/

The students were not asked to count the number of words per language in a tweet to determine the majority of the words in a codeswitched tweet, rather this was left for them to judge. However, as this could be ambiguous in cases where the tweet is equally or almost equally codeswitched, the annotators were given the option to choose *I don't know* to answer the question. For example:

Tweet: *Can't stop watching the promo ?? shu ra7 t3mlo fina bel 7al2a?*
Ambiguous. *I don't know*

Hence, for each tweet the annotators may choose one of three given answers: *yes, no,* or *I don't know*. For example:

Please check whether each tweet is written mostly in Arabizi:

7elo w mesh 7elo	Yes	No	I don't know
live for you not for them	Yes	No	I don't know
keep a 3aj2a kit ma3ak matra7 el first aid kit in case 3le2et bi 3aj2a	Yes	No	I don't know

This should result in a Twitter dataset from Lebanon that is annotated as *Arabizi*, *Not Arabizi*, or *I don't know*. To annotate for sentiment within the same task, if the users annotated a tweet as Arabizi (yes), only then they will be asked about the sentiment of that tweet instantly, with smileys representing *positive, negative, or neutral* and an *I don't know* answers to choose from as well. For example

Please check whether each tweet is written mostly in Arabizi:

7elo w mesh 7elo	Yes	No	I don't know

What is the sentiment of this tweet?

☺ ☹ I don't' know

live for you not for them	Yes	No	I don't know
keep a 3aj2a kit ma3ak matra7 el first aid kit in case 3le2et bi 3aj2a	Yes	No	I don't know

This should extend the annotation of *Arabizi-yes* tweets to *positive, negative, neutral, or I don't know*. The goal is to split the resulting annotated Twitter data into two datasets: Arabizi-yes and Arabizi-no AI dataset and Arabizi positive and negative SA dataset.

We provided each annotator a separate account to login to the platform. We added a timer to record and show the time of the annotation for the users, pause and resume buttons for a better experience and to hold the timer when idle. We added a progress bar meter that displays the percentage of completed tweets to show the annotators how much they have completed and how far they are from reaching the target. We also added results bar meters that display the percentage of Arabizi tweets and their sentiments as they annotate. A screenshot of the platform is presented in Figure 4.2.

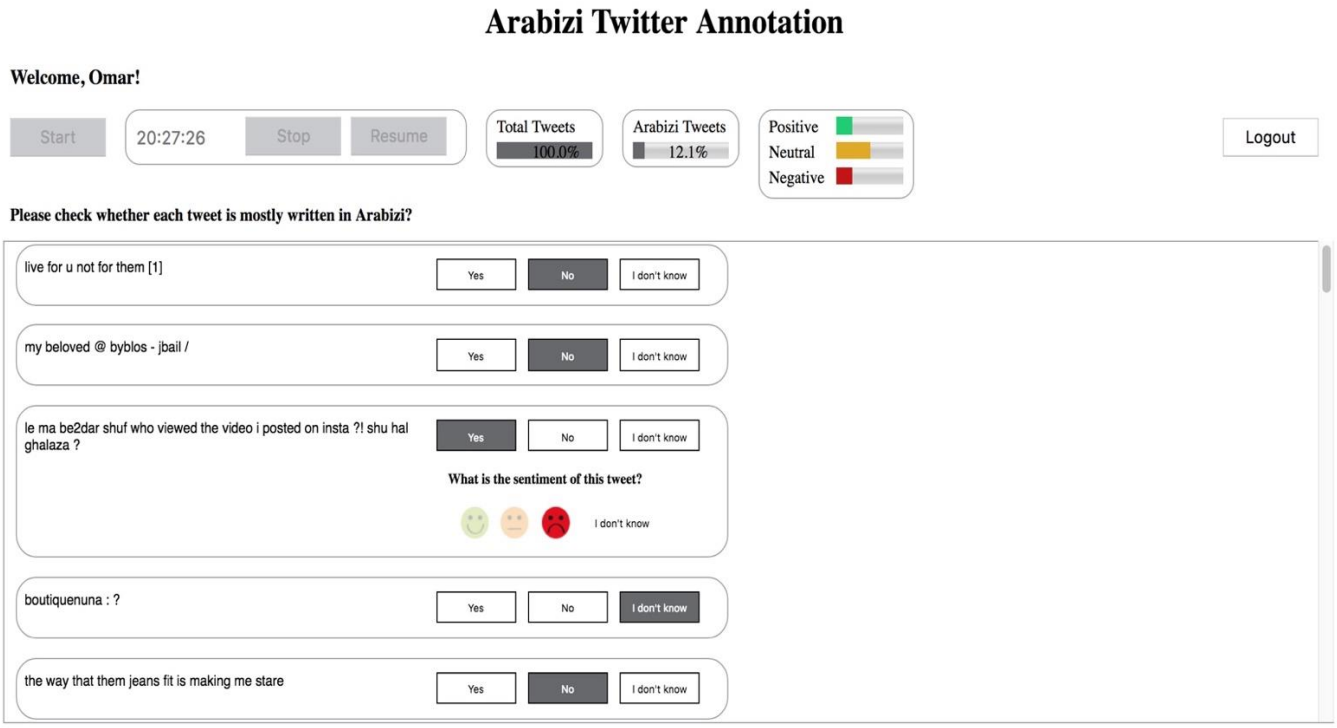


Figure 4.2 Arabizi Twitter Annotation Platform

4.2.3.5 Instructions

As mentioned previously, before annotating the 30K tweets, we assigned the students a preliminary annotation task of 1K tweets to estimate the annotation time and to observe the quality of the annotation. We planned to read the students' annotations to identify any shortcoming, so we may notify the students by showing them where they fell short and guide them further.

We selected 1K tweets randomly from the preprocessed 30K Latin script tweets and loaded them into the annotation platform. We guided the students on how to use the platform and presented them with some annotation examples.

We observed the preliminary annotation to find that each student has a shortcoming. We present below the major shortcomings and how we guided each student afterwards:

1. One student identified tweets as Arabizi based on the first word(s) only, though codeswitching appeared later in the tweet. For example:

Tweet: *Khalas can we fast forward to Christmas ?*
Labeled: Arabizi

We reminded this student that first question, *is the tweet written mostly in Arabizi*, relies on the majority of words in the tweet. As such, reading the entire tweet is required to identify the language of the tweet as Arabizi or not Arabizi.

2. Despite instructing the students to answer *I don't know* for ambiguous tweets, one student reported that they were confused about the sentiment of some tweets. For example:

Tweet: *re7et l beet popcorn* / *house smells like popcorn*
Tweet: *beshfa2 3layon* / *I feel pitty towards them*

We advised this student to judge the sentiment of the tweet based on the impression they get from the tweet. If they were confused whether a tweet is positive, negative, or neutral, we encouraged them to answer *I don't know.*

3. In several cases, all students were not considering the content of the tweet for sentiment, instead, they judged a tweet by the expressions it contained. For example, if there were expressions of laughter *haha, hehehe, lol,* etc they judged the tweet as positive.

Tweet*: mahada lekechoun lal la3ibeh lebneniyeh hahahahha* / *Nobody is looking at the Lebanese players hahahahha.*
Labeled: positive

Tweet*: haha tabashna bl exam / haha we failed the exam*
Labeled: positive

We instructed all students to annotate for sentiment based on the content of the tweet and not to take expressions as key features to identify the tweet as positive or negative without reading the tweet.

After informing each student where and how they could have annotated better, we formulated a list of annotation instructions for the students and re-iterated it onto them with several examples prior to starting the 30K tweets annotation task. The list of instructions is presented in Figure 4.3.

4.2.3.6 Results

Three Lebanese native students annotated 30K preprocessed Latin script tweets from Lebanon. The annotation started in March 2017 and completed in May 2017 at the students' free time and own pace. The annotation was based on two questions:

1. Is the tweet written mostly in Arabizi? (yes, no, I don't know)

   if yes

2. What is the sentiment of the tweet? (positive, negative, neutral, I don't know)

We introduce the annotation results of the first question by presenting the number of each label *Arabizi-yes, Arabizi-no, and I don't know* in total. For example, lets assume three annotators annotated three tweets:

	Annotator	Annotator	Annotator
Tweet	Arabizi-yes	Arabizi-yes	Arabizi-yes
Tweet	Arabizi-yes	Arabizi-no	Arabizi-no
Tweet	Arabizi-yes	Arabizi-yes	Arabizi-no

Table 4.1: Example of total count

Then the total number of the label Arabizi is 6 and the total number of the label non Arabizi is 3.

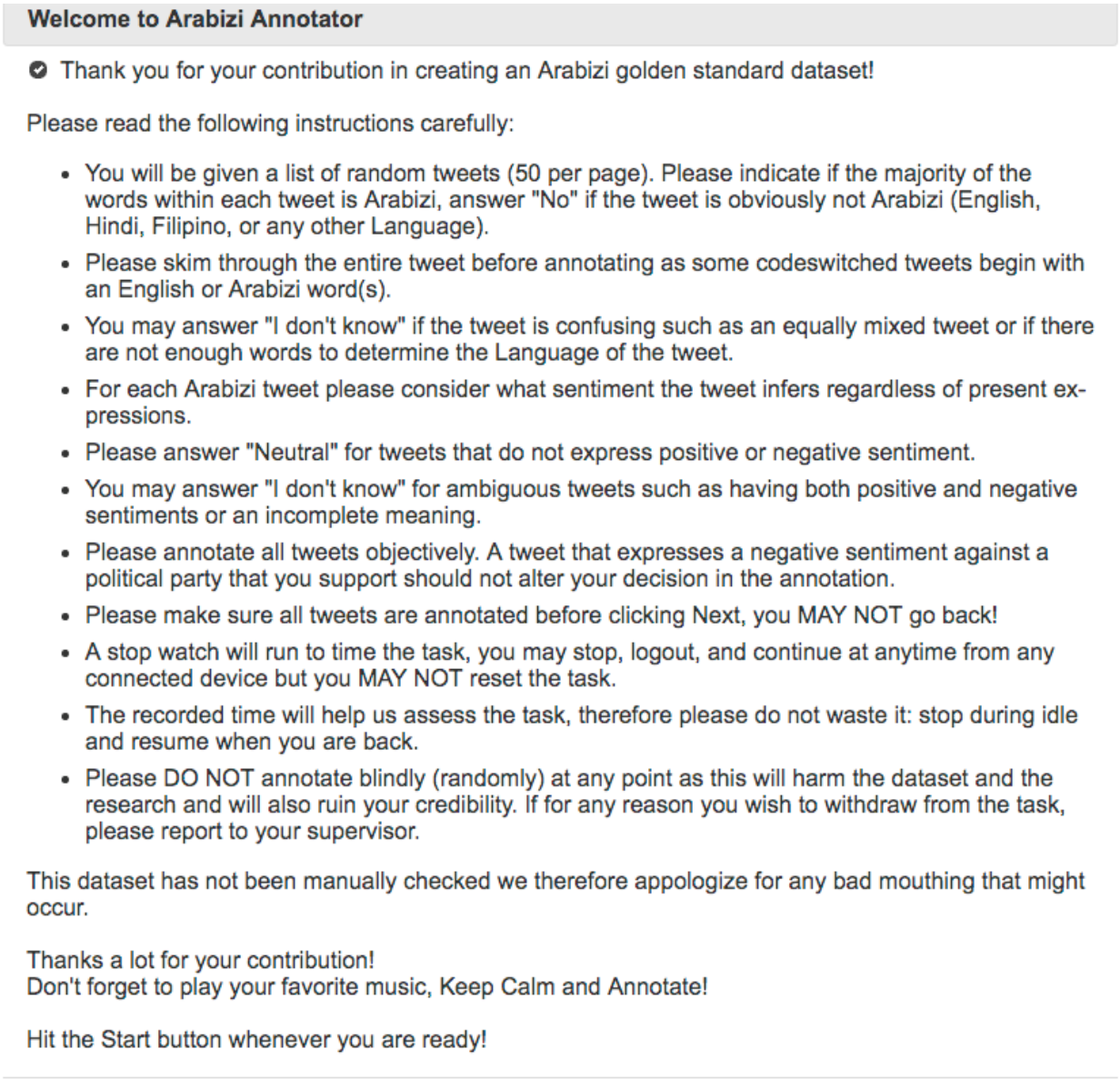


Figure 4.3: Arabizi Annotation Instructions

The total number of labels for the first question is presented in Table 4.2. There were a total of 4.3K *yes*, 27.6K *no*, and 641 *I don't know*. Fleiss Kappa (Fleiss, 1971) was applied to measure the agreement among the students scoring a substantial agreement of 0.74 (Landis and Koch, 1977).

Tweets	Arabizi	Not Arabizi	I don't know	Kappa
30K	4.3K	27.6K	641	0.74

Table 4.2: Arabizi Annotation of 30K Tweets

We present this annotated Twitter data in number of annotator agreement in Table 4.3. For three annotators we have two cases. For example, for the annotation of the first question, whether a tweet is written mostly in Arabizi:

1. Any two annotators label a tweet as Arabizi: 2-annotator agreement.
2. All three annotators label a tweet as Arabizi: 3-annotator agreement.

All Tweets	Arabizi Labelled	Agreement
30K	3.4K	2 Annotators
	2.2K	3 Annotators

Table 4.3: Arabizi Annotator Agreement

Assuming that the higher the value of annotator agreement, the more accurate the annotation is, which is generally the case, there would be a direct relation between the accuracy and the size of the data. Hence, the more accurate the annotation is, the less the number of tweets. The size of the data is critical for the sentiment analysis evaluation; the more data is available the better the evaluation would be (Chapter 3). Given that Arabizi makes up a small percentage of the Twitter data in Lebanon, we chose the 3.4K Arabizi tweets of two annotator agreement for the SA dataset, sacrificing some accuracy for size. As for the AI dataset, we chose the 2.2K Arabizi tweets of three annotator agreement to train and test an Arabizi identifier, assuming that 2.2K Arabizi tweets would suffice to train an Arabizi Language Identifier within a limited number of Latin script languages, hence sacrificing size for accuracy.

We balanced the 2.2K (3-Annotator agreement) Arabizi with another 2.2K (3-Annotoator agreement) randomly selected non-Arabizi tweets to produce the first dataset.

Arabizi Identification (AI) Dataset: 4.4K Latin script tweets (2.2K Arabizi and 2.2K Not Arabizi).

We now move forward with the 3.4K (2-Annotator agreement) Arabizi tweets to present the annotation of the second question. We present the sentiment annotation of these tweets in Table 4.4. Out of the 3.4K tweets there were a total of 1.2K *positive*, 1.4K *negative*, 2.1K *neutral*, and 172 *I don't know*. We note that the Kappa here is impacted by the agreement of the first question. This 3.4K tweets are the result of two or more Annotator agreement from the first question whether the tweet is Arabizi or not, therefore for the majority of the tweets that only two agreed upon, the third annotator had no opinion in the second question about the sentiment of the tweet.

Tweets	Positive	Negative	Neutral	I don't know
3.4K	1.2K	1.4K	2.1K	172

Table 4.4: Sentiment Annotation of 3.4K Tweets

Similarly, we present this annotated Twitter data in number of annotator agreements in Table 4.5.

Arabizi Tweets	Agreement	Sentiment Labelled	Positive	Negative	Neutral	I Don't Know
3.4K	2 Annotators	2.9K	801	881	1.2K	7
	3 Annotators	1.1K	389	363	431	2

Table 4.5: Sentiment Annotation Annotator Agreement

To avoid reducing the size of the dataset, we chose those 2.9K tweets of two annotator agreement to create the SA dataset. The evaluations we present in Chapter 7 are two-class sentiment analysis evaluations, positive and negative only, as such, we do not include the neutral tweets in creating this dataset.

From the 2.9K (2-Annotator agreement) sentiment annotated data, we took 800 positive tweets and balanced them with 800 randomly selected negative tweets to produce the second dataset.

Sentiment Analysis (SA) Dataset: 1.6K Arabizi Tweets (800 positive and 800 negative).

As a result, we have two balanced datasets.

- AI Dataset: 4.4K Latin script tweets (2.2K Arabizi and 2.2K non-Arabizi).
- SA Dataset: 1.6K Arabizi tweets (800 positive and 800 negative).

We use the AI dataset to train an Arabizi Identifier to create a large Arabizi corpus in the next section for lexical expansion described in Chapter 6. We use the SA dataset to evaluate the sentiment analysis approach in Chapter 7.

4.3 Facebook Corpus

After showing how the annotated Arabizi dataset is prerequisite to answer RQ2 in Chapter 7, *How could an Arabizi lexicon be developed and used for Sentiment Analysis*, for evaluating the performance of the proposed sentiment lexicon. We now show how a large Arabizi corpus is prerequisite to answer RQ3 in Chapter 7 as well, *could word-embeddings enhance the performance of Arabizi sentiment analysis*.

A twitter dataset of 3.4K short messages is very small to train word embeddings. This section describes the creation of a corpus composed of 1M Arabizi Facebook comments in four subsections: overview, collection, preprocessing, and identification.

4.3.1 Overview

In Chapter 6 we propose a new Arabizi sentiment lexicon that will be created in two stages:

1. Sentiment Words Generation
2. Lexical Expansion

With the lack of Arabizi lexical resources for the Lebanese dialect, the goal in the first stage of creating the lexicon is to generate Lebanese dialect Arabizi words. However, Arabizi as to Arabic is rich in morphology with an added inconsistent orthography. As mentioned in Chapter 2, these two factors had led to the possibility of having a wide range of forms,

inflectional and orthographic, for most Arabizi words. The richness in inflectional morphology and the inconsistent orthography causes a high degree of lexical sparsity in the text, therefore, in the second stage we try to minimise the degree of the lexical sparsity by encompassing as many forms as possible for every sentiment word into the lexicon.

Since most words could be inflected in different ways in Arabic and each inflection could be spelled differently in Arabizi as explained in Chapter 2, we chose to expand the proposed sentiment lexicon by finding forms of sentiment words that are written naturally in text as opposed to hand crafting a far-fetched rule-based inflection and orthographic generator.

We propose to use word-embeddings, a neural network based architecture, to retrieve forms of the sentiment words that are inflected or naturally written differently. The idea of using word embeddings to discover the inflectional forms and orthographic variants of the sentiment words is motivated from the word similarity application of word embeddings.

The notion of word similarity is to input one-hot encoded vector of the words in a vocabulary into hidden layers of neural network that finds relations among these words and outputs them as vector representations, vectors of real numbers. Each output word vector is composed of the probabilities of another word appearing next to or before it, among other numbers. After the neural network represents each word in the vocabulary in a dependency vector, word similarity could be calculated through the similarity of the vectors. Words of similar vectors should be similar.

An example of retrieving nearest neighbours for the word *apple* using word2vec.

Apple: almond, cherry, plum, macintosh.

In this case the word neighbours are similar to the input word *apple* in meaning and semantics but not in syntax. (Mikolov, et al., 2013) mentions in the word2vec paper that this model can capture syntactic similarities as well such as:

Slow: slowly
quick: quickly

However, Arabic is rich in morphology and Arabizi vocabulary is highly sparse in terms of syntax, we therefore desire to discover morphologic and orthographic variants using this approach such that:

Desired morphologic retrieval:
7ob / love: 7abibi, 7aboub, ma7boub, 7abibte, bet7ib, be7ebak, b7ebkon, 7abeit, 7abeita, 7abaytak, 7abaytik, 7abayton, etc.. / my-love, loved-one, loved-one (another form), *my-love* (feminine), *you-love, he-loves-you*(masculine), *I-love-you* (plural), *I-loved, I-loved-it, I-loved-you* (masculine), *I-loved-you* (feminine), I-loved-them

Desired orthographic variances retrieval:
7abibi / my-love: *7bb, 7abb, 7bbi, hbbi, hbb, habibi, habeebi, habibiiii, habeeeebeee, habbb, hbbb, 7bbb* etc..

In order to find out, we need to train word embeddings on a large Arabizi corpus. The corpus used in the previous word2vec example (Mikolov, et al., 2013) is composed of 30 billion words.

4.3.2 Collection

The incentive of compiling a corpus is to discover Arabizi word forms and variances in their natural orthographies, the way people write them, because the Arabizi orthography is inconsistent. That being said, a large number of Arabizi text is needed to maximise the chance of discovering such words. For the case of creating annotated datasets, Twitter was a good source, for it is public, contains Arabizi, and its messages are short. We believed that since a short message (220 character) is more likely to focus on a single topic, it would be more suitable for Arabizi and sentiment annotation than long messages such as a paragraph. For word embeddings we needed a large amount of Arabizi messages, regardless of their size. Unlike collecting Arabizi tweets from Twitter by streaming live tweets from Lebanon, where Arabizi comprise 9.3% of the data, we take a different approach for creating the corpus, we collect comments that have been posted already in public Facebook pages.

Facebook, the famous social network, where users connect with each other, post texts, images, and videos that are subject to reactions and comments from users within their

network of friends. It also contains public pages that could be run by organisations such as artists' fans, political groups, news agencies, community shows and groups, comedy shows, etc. The publicity of such pages makes the content of the pages accessible by anyone. The content in these pages is posted by the organisations or individuals who run the pages, text, images, videos, or events, with an open space for any user, usually people who follow the page, to express their opinion by reacting, commenting and engaging in conversations on these posts.

We create the corpus by collecting all public Latin script content and comments and that are posted in response to the content, from a list of public pages from Lebanon. We select pages based on the following criteria:

1. Popular
2. Active
3. Lebanese audience
4. Arabizi comments

Popular, at least having a couple of thousand follower ensuring that the page is not limited to small number of people. Active, where followers of the page comment and interact with the posts. Some pages do not receive comments to their posts. Lebanese audience, a page from Lebanon does not necessarily indicate that the followers are from Lebanon, this could be identified through manual observation from the dialect of the comments and usernames. Arabizi comments, pages where Arabizi is regularly used in the comments. In Figures 4.4 and 4.5 we present snapshots from the comments section of random posts from four different public Facebook pages in Lebanon.

As can be seen from Figures 4.4 and 4.5, the languages of the comments are different in different pages, though from Lebanon, however consistent per page. Each of these pages contain similar comments, in Language, under the rest of their posts, i.e., just like the example from the first page, where Arabizi and English comments are present in that post, we observed the comments are Arabizi and English in the rest of the posts. The commentators in pages 2 and 3 comment in Lebanese dialect Arabic and English respectively. As for the 4th page, majority of the comments are of different Arabic dialects such as Gulf Arabic and Egyptian although the Artist is Lebanese. As such, among these four pages, only the first one would serve the purpose of creating a Lebanese Arabizi corpus.

We manually scouted Facebook pages from Lebanon to select a list of pages that match the criteria that we have set to extract the textual data. We found the pages using Social Bakers[22] and Facebook's Top Pages suggestions.

Social Bakers is a social media statistics webpage that provides lists of most popular pages on social media per country, among Facebook and others. We skimmed through each of the popular Lebanese pages checking if the page is active and whether Arabizi is apparent in the comments section. We followed each of these pages. Facebook then started to suggest similar pages that also matched our criteria. In total, we selected 49 pages of various genres.

We wrote a script that calls Facebook API to iterate over all posts (texts, images, videos, and events) posted in a public page and extract all comments and replies from each of these posts. The script collects all Latin script text from the posts, skipping Arabic comments, post by post, sequentially up to the very first post posted by the page. We launched the script over the selected 49 pages in 2017 harvesting around 2.2M Latin script comments. The list of the pages is presented in Table 4.6 along with some statistics including the number of comments collected from each page.

4.3.3 Preprocessing

The purpose for creating an Arabizi corpus is to retrieve inflectional and orthographic forms of input words, sentiment words in our case. For that, we want to filter the collected Facebook data from comments that are not composed of words. Similar to the previous preprocessing applied on the Twitter data in Section 4.2.2, we also use regular expressions to remove URLs, hashtags (words preceded by #), mentions (words preceded by @), media attachments [image attached or video attached] and non-ASC characters. Similar to the Twitter data, filtering comments from hashtags, mentions, URLs, media attachments, and non-ASC characters resulted in many comments lacking an alphabet as well. We removed all such comments. This reduced the comments from 2.2M to 2.1M.

[22] https://www.socialbakers.com/statistics/facebook/pages/total/lebanon

1. El 3ama: Community/Comedy
Active
500K followers
Arabizi comments

2. Aljadeed Online: Local and International News
Active
4M followers
Lebanese Arabic comments – No Arabizi

Figure 4.4: Arabizi in public Facebook pages from Lebanon

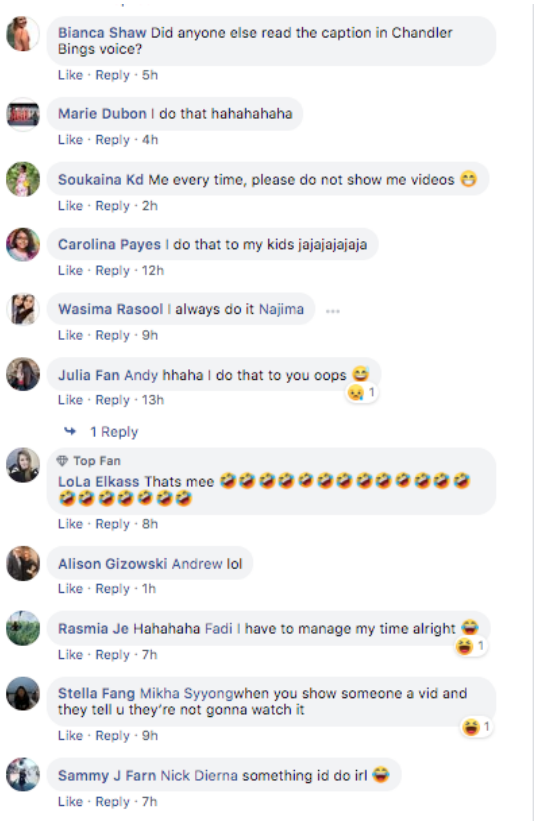
4.3.4 Arabizi Identification

The 2.1M comments harvested from the mentioned public Facebook pages, that we observed to contain Arabizi within their comments, consist of Latin script languages. Since we filtered any comment written in Arabic script as we collected the comments, the apparent languages in the current collection are English and Arabizi. At this stage we would like to identify the Arabizi comments automatically to create an Arabizi corpus. Identifying the language of multi-lingual text is a Language Identification task. Below are examples in our case:

Comment*: Ra7 tnawer trablos ahdam 3alam*
Automatic Identification: Arabizi

Comment: *he mentioned the difference between the Jordanian and Lebanese Salam exactly how it happens between us always*

Automatic Identification: English



3. Virgin Radio Lebanon: Media

Active

13M followers

English comments – No Arabizi

4. Elissa: Artist/Fan page

Active

22M followers

Mixed Arabic dialect comments

Figure 4.5: Arabizi in public Facebook pages from Lebanon

Page	Genre	Since	Followers	Comments
Adel Karam	Talk Show / Sarcasm	2011	2M	24.9K
Ahmar Blkhat El3arid	TV Show / Society Problems	2011	2.5M	125K
CHiNN	TV Show / Sarcasm	2011	165K	41K
Helem Lebanon	LGBTQ	2010	13K	4K
Jeandarc Zarazir	Comedy	2017	70K	271
Lebanese Army	Government	2014	270K	554
Lebanese Memes	Memes / Sarcasm	2012	460K	101K
Lebanon Files	News	2010	440K	199.8K

Lebanon on my Mind	Blog	2016	190K	16.2K
MEA	Airline	2011	386K	41.7K
Merheb Simi	Comedian / Internet	2015	65K	14K
Micheal Aoun	President / Politics	2013	196K	94
The Shock Leb	Community	2017	217K	2.6K
Wissam Doc Comedy	Standup Comedian	2012	50K	14K
Wizz Fun Leb	Comedy / Internet	2017	13.7K	251
Ayam Serious	Comedy / Internet	2016	72K	8.6K
Roger Baz	Comedian / Internet	2014	45K	13.4K
BBChi News	TV Show / Sarcasm	2016	156K	25.3K
Bint Jbeil	Local News	2010	5.1M	145K
Buzz Vodka Mix	Memes	2012	32K	8.7K
Farixtube	Comedian / Internet	2016	30K	11.4K
Hicham Official Page	Talk Show / Sarcasm	2010	513K	26.4K
How About Beirut	Pranks	2013	1.6M	15.4K
How I Take my Coffee	Comedy / Internet	2015	21K	9.5K
Just Edhak	Comedy / Internet	2012	200K	13.2K
Kawalees Beirut	Comedy / Sarcasm	2015	25K	19.9K
Lahon w Bas	Community / Talk Show	2015	847K	57K
Lebanese Forces	Political Party	2010	357K	280.8K
Lebnani Bloc	Talk Show / Politics	2015	75K	3K
Marroun Azzi	Humanitarian Support	2017	15K	342
Mawtoura	Sarcasm	2014	174K	25K
MTV Lebanon	TV Channel	2010	5.2M	324K
Oh my Jad	Comedian / Musician	2010	107K	9.8K
Pierre Hachache	Sarcasm / Politics	2008	173K	29.3K
Quickies Leb	Standup Comedy / Band	2016	120K	35.7K
Samy Gemayel	Political Figure	2010	285K	54.2K
Sarah Abi Kanaan	Actress / Fan Page	2014	52K	1.6K
Shroud w Tahwaji	Comedy / Internet	2015	6K	326
Stepfeed Lebanon	Community / Arab News	2016	27K	18.2K
Stop Cultural Terrorism	Community	2011	80K	59.8K
Tayyar	Political Party	2009	813K	321.5K
Cheyef 7alak	Community / Anti-Racisim	2011	49K	12K
Eich w Kol Ghayra	TV Show / Pranks	2016	107K	23.2K
El 3ama	Comedian / Sarcasm	2016	537K	37.6K
Mukhtar007	Comedian / Internet	2016	537K	8.7K
You Stink Lebanon	Anti-Corruption	2015	280K	85.5K
Beirut Madinati	Political Group	2016	68K	8.8K
Wen el Dawle	Exposing Corruption	2017	355K	102K
Zaatar w Zeit	Restaurant	2009	765K	49.3K

Table 4.6: List of Facebook Pages

Classifying text as Arabizi or Not Arabizi is a two class, binary, classification problem. Since we have a dataset of tweets that are manually annotated as Arabizi-yes and Arabizi-no, we train a ML classifier to classify the Facebook comments as Arabizi or Not Arabizi.

ML classifiers learn from labelled data, learn from example in other words. A classifier learns the patterns in the data that makes the data belong to a certain class until it reaches a

level where it is capable of relating the learned patterns with new unlabeled data to predict a class for it. This data however is presented in numbers for the ML algorithm to learn patterns from. The ML classifier converts the text into vectors of real numbers, a process called vectorization. We test two classifiers that are known to perform well in binary classification: Support Vectors Machine (SVM) and Logistic Regression.

SVM vectorises the training examples and plots them into a graph. It then finds a hyperplane that splits the two different classes from each other while maximizing the margin of the split. A small visualization of this process is presented in Figure 4.6.

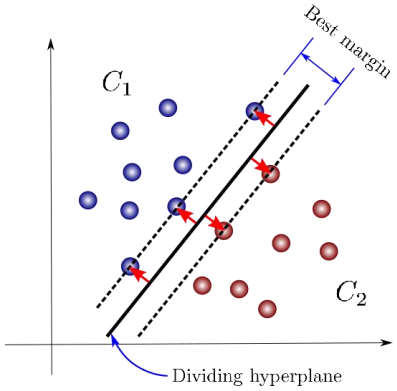


Figure 4.6: Support Vector Machines

After the split the classifier becomes capable of predicting a class for the new incoming vectors of sentences.

Logistic Regression vectorises the training examples and plots them into a graph as well. It then predicts the probability of an input vector belonging to a class by fitting the training data to a logit (sigmoid) function. A small visualization of this process is presented in Figure 4.7.

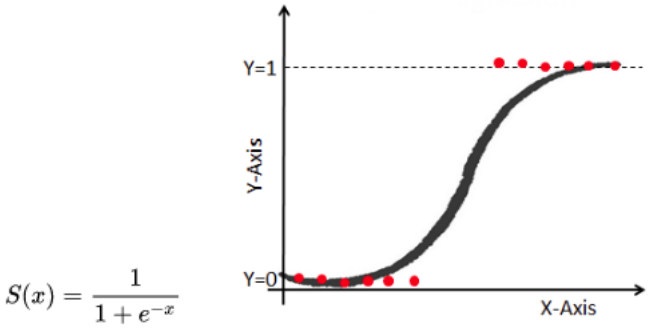


Figure 4.7: Logistic Regression

A threshold boundary is then set to classify input vectors into one of the classes, above or below the threshold.

We used the AI dataset from Section 4.2 to train both classifiers. The AI dataset consists of 4.4K Tweets (2.2K Arabizi and 2.2K non-Arabizi) with a three annotator agreement for both classes. We used the unigram feature for both classifiers. The unigram is the occurrence of every word in the vocabulary of the training data, hence the words are presented in vectors from a bag of words. The occurrence of the word in the bag of words, vocabulary of the training data, is the only feature that the classifier learns patterns from regardless of the frequency or the position (co-occurrence with other words) of the words. An example[23] of the unigram feature vectorisation is presented below.

If the following sentences make the training data:

There used to be Stone Age
There used to be Bronze Age
There used to be Iron Age
There was Age of Revolution
Now it is Digital Age

The vocabulary of the training data would be

[23] Example taken from: https://medium.com/@paritosh_30025/natural-language-processing-text-data-vectorization-af2520529cf7

There, was, to, be, used, Stone, Bronze, Iron, Revolution, Digital, Age, of, Now, it, is

A unigram feature, occurrence of word in a sentence, vectors would like:

There used to be bronze age = [1,0,1,1,1,0,1,0,0,0,1,0,0,0,0]
There used to be iron age = [1,0,1,1,1,0,0,1,0,0,1,0,0,0,0]
There was age of revolution = [1,1,0,0,0,0,0,0,1,0,1,1,0,0,0]
Now its digital Age = [0,0,0,0,0,0,0,0,0,1,1,0,1,1,1]

Training and testing a ML classifier is usually done by splitting the labelled dataset into training and testing data, where the classifier learns from the labels in the training data to classify the testing data. The performance of the classification is then validated against the correctness of the testing data. The data is usually split in the following fashion, a large part for training and a small part for testing, as in Figure 4.8.

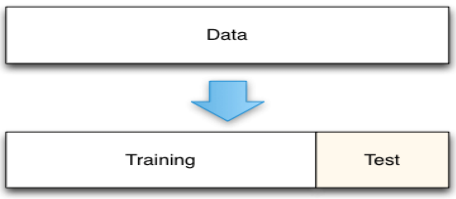


Figure 4.8: Training and Testing Data Split

However, our dataset is composed of 4.4K tweets, a relatively small dataset for classification, thus using this approach risks the possibility of missing important patterns from the text that was not used for training. As such, we use k-fold cross validation technique where the data is split into training and testing k-times to ensure that every pattern in the dataset has the chance to appear at least once in the training and testing parts. Finally, the performance of the classifier in classifying the testing data within each fold will be averaged. A 5-fold cross validation visual is presented in Figure 4.9.

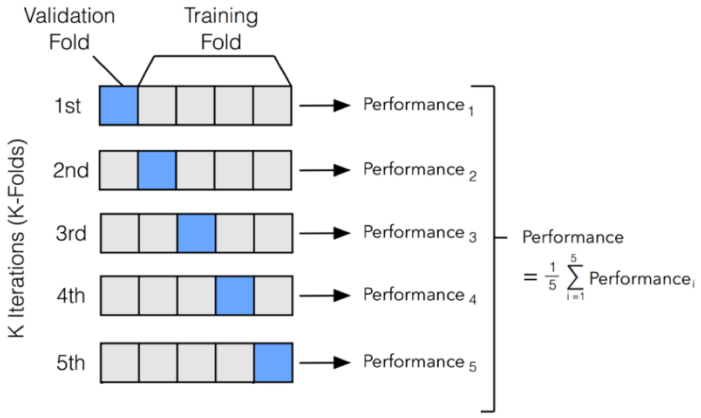


Figure 4.9: K-Fold Cross Validation

Following this approach, we shuffled the AI dataset and split it into 10 folds for cross validation, we trained and tested SVM and Logistic Regression classifiers for the two class Arabizi / Not Arabizi classification using the unigram as the input feature. The results of both classifiers against the AI dataset are presented in Table 4.7:

SVM performed slightly better than Logistic Regression by a negligible difference. We therefore chose SVM to identify the Arabizi comments from the 2.1M Latin script Facebook comments obtaining a corpus of 1M Arabizi comments. This shows that Arabizi to non-Arabizi is almost equal in Latin script texting in the selected pages on average. A snapshot of some comments classified as Arabizi (1) and non-Arabizi (0) is presented in Figure 4.10.

4.4K Tweets: 2.2K Arabizi – 2.2K non-Arabizi

Classifier	Recall	Precision	F1-Score	Accuracy
SVM	0.93	0.97	0.95	0.95
Logistic Regression	0.93	0.96	0.94	0.94

Table 4.7: Arabizi Identification

We will use this corpus to train a word embedding space in Chapter 6 to discover inflectional and orthographic forms of input sentiment words as an expansion of the proposed lexicon in Chapter 5.

Facebook Comments	Arabizi
shu 2smo	1
Milia antoun knt 3m b2ra comments elk ente ktr 8abye	1
Sevan hahahahahahahahhaa	0
Mafi chocolat?	1
Bs mourada2 nafsie:)	1
Chou hal titre l da3eche .... chwey chwey alayna	1
Bravo 3leyk bado aktar men hek	1
w iza 5alaste M1 ra7 yeje el naizak abel ma t5alse el M2 😂	1
Fedaa Shall shu ma kenit asbeba hy lezim ba3ed el 7al2a tet7awal lal kada2 a...	1
Ma kel balewena ma lfalastiniye	1
One day... is too vague! We need it soon!	0
Wonderful 💜	0
Musa Fahme	0
Ma hek hehehe😂😂	1
Hahahahahhahahahahahahha tb skete	0
Talama el nasa7a jamel ya bnteee 5ale el kinder se7r jamelik 😉 yala 5abtine...	1
Fe kamen hotel l movenpick 3am ya3mel nafs l gayme bass sawda	1
byoktori3 bie jhanam hal sefeh wariban ebno rah yelha2o 3a jhanam	1
Indeed	0
Bnesbe lal de3ich li 3am ytfalsaf ya 5aye chou 5asak enta fiya aw bi aya bent...	1
kafa ba2aaaa harrammmm ba2aaa 2erfet al 3allammm ma ba2a tehmol al 2os...	1
Yalla il 3ad al 3aksi	1
ah ya habiler	1
Ro7 kis 2i5tk 2a5o manyoki ya 2ibn l sharmota	1
Nice work...	0
hhhhhhhh 3a l a7jemm :p	1
guck mal die harqm	0
Ze3rane	0
Penguin 😂	0
Such a shame ! Sorry for your loss imad !	0
L sha2ra metel rita 😂😂😂	1
7eke badre :p	1
#NoFlyZone4Rojava	0
🐸wen el 3aniffa ma fhemet ma adrin 3am yt7arako 🐸3am ywalwlo bas	1
Kawtharani AhmAd	0
eh defe3 3an tizak w ni3ak bado yintek emkon ya mojrmin	1
Alf sahtain!Ta3mina ma3ak!	1
Yet ksar idioun nchalah	1
eno ma yekhedon 3a dafe3teen	1
Ma 7ada bimout wara meyto bala 3arek	1
#mtvlebanon	0
Allah yerhamak ya batal!!	1
D3a lmo7rbet lflstnye wd3wto kanet 3a 7a2 ya ret ma 5alo wla flsteni ma kan l...	1
Yyiii haraaam 3emlo accident men waranaaa ... ma hada allon yettalllaa3oooo	1
El ya 3antar min 3antarak ? Jewab t3antaret w ma 7ada radné	1
Allah yekhedoun	1
The faghters should be burn in jail	0
Congratulation all and thank you Zaatar W Zei	0
Kil lebneniyin yash3ouron metlik bil hezen	1
Eh la2eno baya ba3ata la 3nd dr.Nader feshil bihadaf 2ena tmout w yshawehl...	1
Noutalib b tar7iloun aw b tajdid l 2ikame lal lebnaniye hahahaha.wtf	1
Hahaha dadsss also and friends 😂 @nadimkhairallah sooo u bitch	0
Hhhhhhhhh kl ma ashofn atzkrk hhhh a7rjn 5tea 😂😂	1
Frequency:12360 MHz	0

Figure 4.10: Arabizi Identification Examples

4.4 Discussion

In this chapter we created annotated Arabizi Twitter datasets to train an Arabizi Identification classifier and to evaluate the sentiment analysis approaches in the next chapters. We also

created an Arabizi Facebook corpus to expand the proposed sentiment lexicon for a wider coverage of sentiment words.

We list some of the limitations we faced in creating the datasets and the corpus below:

1. Time: It took us three months to stream 177K tweets. The streaming rate was influenced by several factors:
   a. Location information: We suspect that many tweets were not collected by the stream API, as Twitter provides its users the option to disable the location information from their tweets.
   b. Technical issues: Streaming tweets is not as smooth as collecting tweets that have already been posted and stored in Twitter's database. Using Twitter API, the stream was interrupted[24] several times. We had to keep an eye on the running script and restart it whenever interrupted.

   It also took 30 hours to annotate 30K Tweets for Arabizi and sentiment. The annotation was at the students' pace having long periods of time intervals whenever they had to meet an academic responsibility or personal circumstance.

   Additionaly, after observing that the Twitter API was not accurate in identifying the language of the tweets, mainly because of the presence of non-alphabet text or codeswitching, we decided that the students should infer whether a tweet is Arabizi or not to maximise the quality of the annotations. However, looking back at this, we realise that we could have saved time and annotated more Arabizi tweets had we filtered out the English tweets using an external language identification library such as Google API[25] after cleaning them from non-alphabet text. In any case, we did not foresee that 30 hours of annotation would span over 3 months at the students' comfort.

[24] Twitter stream interruption is a common issue listed by Twitter API https://developer.twitter.com/en/docs/tutorials/consuming-streaming-data
[25] https://cloud.google.com/translate/docs/basic/detecting-language

2. Annotation agreement: In total there were 801 positive and 881 negative tweets where two students agreed. The annotation for the second question *what is the sentiment of the tweet?* depends on the answer of the first question *is the tweet mostly Arabizi?* as such if one student disagrees with the rest in the first question they will decrease the chance of two annotator agreement in the second question. We present and analyse some tweets where not all three students agreed upon from both questions:

   Tweet: *bhebbik ya ashta / love you (oh ashta)*
   Ambiguous: this phrase is used for sarcasm

   Tweet: *ent btestehal / you deserve it*
   Ambiguous: could be good or bad

   Tweet: *her nails ktir helwin / her nails are so nice*
   Codeswitched: Depends whether the student considered this as Arabizi or not.

   Tweet: *enta rteh ma 3lek / you rest don't bother*
   Ambiguous: could mean *none of your business* or *don't worry about it*

   Tweet: *Good evening ya a7la 3arous / Good evening oh prettiest bride*
   Codeswitched: Depends whether the student considered this as Arabizi or not

   Tweet: *leh fi 3alam bta3mel copy paste la nekat mn facebook aa twitter / why do people copy paste jokes from Facebook onto Twitter*
   Ambiguous: could be expressing anger or just asking a question

   Tweet: *khayye khalas. Get over highschool. Walaww / bro enough. Get over highschool. Commonn*
   Codeswitched: Depends whether the student considered this as Arabizi or not

   Tweet: *oumo ba2a /* an expression of negative surprise used in response to something exaggerated such as *common!*
   Ambiguous: Could be slightly negative

As observed from the examples, codeswitching, insufficient information, and ambiguous meanings were reasons for impacting the annotation agreement.

3. Data genre: We harvested 49 public Lebanese pages on Facebook to create an Arabizi corpus. We manually checked the pages against certain criteria, amongst which, must contain Arabizi comments. Although we desire to have had a balanced number of page genres, news, sports, comedy, politics, etc., based on our observation, Arabizi occurs more in pages about comedy and sarcasm. It is less frequent in conversations about news and politics. As a result, the comedy and sarcasm genre comprise around half of the selected pages.

On another front, we found the 1M Facebook comment corpus to contain to 892K unique words, that is almost a unique word per comment, which shows how large the lexical sparsity is in Arabizi.

Nevertheless, to the best of our knowledge, the datasets and the corpus created in this chapter are the first Lebanese Arabizi data resources for NLP. We made all the data resources created as part of this course of research public and free for academic and research use on the project webpage[26].

4.5 Chapter Summary

In this chapter we mined two social media resources to build Arabizi datasets. We collected and preprocessed Twitter data and assigned an annotation task to three students that resulted in two annotated datasets:

1. AI (Arabizi Identification) Dataset: 4.4K Tweets: 2.2K Arabizi and 2.2K non-Arabizi
2. SA (Sentiment Analysis) Dataset: 1.6K Tweets: 800 positive and 800 negative

[26] https://tahatobaili.github.io/project-rbz/

We collected and preprocessed the Latin script comments from 49 public Facebook pages. We then used the AI dataset to train an Arabizi identifier to identify the Arabizi comments from comments written in other Latin script languages. The Arabizi identifier identified 1M Arabizi comments.

5 SenZi: The Arabizi Sentiment Lexicon

أبى فاء الى الفيافى فاذا فاء الفىء فاء

In this chapter we present the core resource of the thesis, SenZi, a new sentiment lexicon for Lebanese dialect Arabizi. We start by addressing RQ2:

How could a sentiment lexicon be developed and used for Arabizi sentiment analysis?

In this thesis, our primary focus is on the design, generation and application of sentiment lexicons for Arabizi. In Chapter 6, we propose expansion techniques for SenZi to increase its coverage. In Chapter 7, we evaluate the resulting lexicon and its expansion.

Lexicon-based approaches for sentiment analysis classify input text into sentiment classes based on occurrences of the lexicon words within the text. Factors like word coverage and polarity scores assigned to the words may influence the performance of such approaches, since lexicons can sometimes have a wide coverage for a particular domain but not for others. On the other hand, lexicon based approaches do not depend on training, or labelled data, which is very expensive to develop.

There are several ways in which the sentiment scores of the words, or the sentiment polarity in our case (positive, negative) can be combined to compute the overall sentiment of the text. These methods take into account: The Part of Speech (POS) of the sentiment words, the position of words within the sentence, or the semantic concepts in the text such as entities and their relations.

In our case, we are considering the classical lexicon-based approach for sentiment analysis where the polarity of terms found in the text is averaged to compute the overall sentiment of the text. We present two examples from the dataset below, assuming that the lexicon contains the following sentiment words:

Ya 7abibetna enti sourtik bi albna wayn ma tkouni
our-darling your picture is in ourheart wherever you are

(*our-darling*: +1) *your picture is in our heart wherever you are*
Total Score: +1
Class: Positive

3alam we27a bada tdhak 3layna
impudent people they want to con us

(*impudent*: -1) *people they want to* (*con*: -1) *us*
Total Score: -2
Class: Negative

The lack of NLP resources for Levantine dialect Arabic in general and Lebanese in specific motivated us to create a new sentiment lexicon for the Lebanese Arabizi.

We start by briefly describing the structure of some known sentiment lexicons in the literature of English and Arabic NLP. We then move on to the design of SenZi.

English Lexicons:

1. Lu Hiu and Bing Liu (Hu & Liu, 2004): A sentiment lexicon consisting of two lists of words, a positive and a negative list. No polarity scores for the words.
2. MPQA (Wilson, Wiebe, & Hoffmann, 2005): A list of words labelled as strong or weak subjective, POS, positive or negative.
3. SentiWordNet (Esuli & Sebastiani, 2007): A list of words containing a positive and a negative score, synsets (one or more synonyms), and glosses.

Arabic MSA:

1. ArSenl (Badaro, et al., 2014): A list of words containing POS, positive and a negative score, and a confidence score.
2. SLSA (Eskander & Rambow, 2015): A list of words containing a positive and a negative score, English glosses, and an objectivity score.

Many efforts in creating sentiment lexicons for Arabic focus on the standard Arabic MSA in the literature of Arabic NLP (Chapter 3). Recently, some dialects are getting resourced such as Egyptian, North African, and Saudi Arabic (Chapter 3). However, there is a severe lack of lexical resources for Levantine Arabic in general and Lebanese in specific. To create a sentiment lexicon for Lebanese Arabizi, we are faced with the challenge of finding Lebanese sentiment words. As this is being our main challenge at that moment, we focused our efforts on finding Lebanese sentiment words, as such we planned to create SenZi, as simple as possible, containing two lists of words, positive and negative, similar to the Hiu and Liu mentioned above.

A new sentiment lexicon that is capable of achieving good sentiment analysis results for the low-resourced Lebanese Arabizi may be later extended to contain sentiment scores per word for improving the analysis accuracy.

As shown in the previous classification example, one way of using a simple sentiment lexicon, as the proposed SenZi, in a lexicon-based approach is to score the positive words +1, and the negative words -1 in the input sentences and sum the scores at the end.

In any case, we plan for SenZi to contain two lists of Lebanese dialect sentiment words, positive and negative. We plan to add any sentiment word we find to SenZi without being restricted to a specific domain.

We build SenZi in two stages:

1. Lexical Generation
2. Lexical Expansion

In the first stage we present the pipeline for generating Lebanese Arabizi sentiment words to create SenZi. In the second stage we expand SenZi, in Chapter 6, by retrieving inflectional and orthographic forms for every sentiment word. Finally, we present a lexicon-based sentiment analysis evaluation of the proposed SenZi in Chapter 7.

5.1 Lexical Generation

Given the severe scarcity of Lebanese dialect lexical resources, our conception of creating SenZi is to handcraft a Lebanese dialect sentiment lexicon of positive and negative words in the first stage and expand it automatically using word embeddings in the second stage. The term expand in this context refers to retrieving inflectional and orthographic forms of the original sentiment words.

Dialectal Arabic (DA) is a spoken Arabic differing among regions (Chapter 2), thus an orthography for DA has not been standardised. This did not prevent Arabic social media users from transcribing their spoken dialect by spelling words using their personal spelling interpretation of spoken Arabic, not following a standard orthography. Lebanese dialect, a member of the Levantine dialect family, consists of many foreign words, however, majority of the dialectal words originate from MSA though could be inflected dialectally or have different meaning. Table 5.1 shows an example of a dialectal positive and a negative word derived from neutral MSA words.

MSA	Lebanese	Dialectal Inflection
Young donkey **جحش**	Stubborn / stupid	Act of being stubborn تجحيش
Digest **هضم**	Digest	Cute / funny مهضوم

Table 5.1: Examples of dialectal meaning and inflection

As such, lists of MSA words could be useful in selecting words that are used in Lebanese dialect Arabic, however, we designed SenZi to be a new simple lexicon composed strictly of positive and negative words, therefore the MSA lists that we can utilise for this task should be sentiment lists. Based on these requirements, we chose relevant lexical resources to transform them into Lebanese Arabizi through a series of automatic translation and manual selection and transliteration. We describe this pipeline in the following subsections.

5.1.1 Overview

We present and brief the architecture of the first phase of SenZi (Generation), then detail every step in the following subsections, and finally end with a small discussion. We design

the pipeline into four stages: Resources, Translation, Selection, and Transliteration, as shown in Figure 5.1.

1. Resources: We chose relevant English sentiment lexicons and a Lebanese word list. We combine the English lexicons to prepare them for translation.

2. Translation: We translated the combined English lexicons to MSA and included synonyms automatically.

3. Selection: We designed an annotation process that involves students selecting relevant words from the Lebanese word list and the translated English sentiment lexicons. We combined the resulting selection from both resources.

4. Transliteration: We manually transliterated the compiled Lebanese Arabic sentiment lexicon into Arabizi script.

5.1.2 Resources

We used two English sentiment lexicons and one Lebanese Arabic word list as the building seeds of SenZi:

1. Hu and Liu[27]: An English sentiment lexicon created by Minqing Hu and Bing Liu, it is composed of 2K positive and 4.8K negative words (Hu & Liu, 2004).

2. MPQA[28]: Multi-Perspective Question Answering subjectivity lexicon created by Theresa Wilson, Janyce Wiebe, and Paul Hoffmann (Wilson, et al., 2005). It is part of the OpinionFinder System that has been developed by the Universities of Pittsburgh, Cornell, and Utah. It consists of 2.7K positive and 4.9K negative words, where majority of the words were collected from (Riloff, et al., 2003).

[27] https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html
[28] https://mpqa.cs.pitt.edu/lexicons/subj_lexicon/

3. Living Arabic[29]: A list of Lebanese dialect words, it is an underlying list of the Living Arabic, or Lughatuna, project developed by Arabic linguist Hossam Abouzahr. Living

   Arabic contains a multi-dialect online dictionary that aims to bridge different Arabic dialects with MSA. It reflects the intensive efforts that Hossam had placed in collecting different dialectal terms from his own research on Arabic dialectology and several resources such as a Lebanese lexicon called معجم الالفاظ اللبنانية by *Anis Freyha, The Olive Tree,* a Palestinian dictionary, and *Syntax of Spoken Arabic by Kristen Brustad*. The list we used is comprised of 7.1K Lebanese Arabic words.

As can be seen in Figure 5.1, our approach to generate Arabizi sentiment words starts from English sentiment words, online translation to Arabic, then transliteration to Arabizi. The LivingArabic word list remains intact till the selection step. We express the rational for translating English sentiment lexicons instead of utilising public MSA lexicons below:

1. Annotation cost: The chosen Hu and Liu and MPQA sentiment lexicons are popular lexical compilations in the literature. They consist of 6.8K and 7.6K sentiment words that are split into lists of positive and negative words explicitly. The mentioned MSA lexicons, ArSenl and SLSA, consist of 28.7K and 34.8K words of all polarities with a high number of neutral words that are irrelevant to SenZi. Since we planned a dialectal words selection step to find the Lebanese Arabic words, it would be quicker to go through a translated Hu and Liu and MPQA over ArSenl and SLSA.

2. Maximise Lebanese Arabic: ArSenl and SLSA are MSA exclusive lexicons. The translation we obtained from translating Hu and Liu and MPQA online indicates that it is not exclusive to MSA rather often contains dialectal Arabic words. We demonstrate this by selecting some translated sentiment dialectal words and checking whether they exist in ArSenl and SLSA, presented in Table 5.2. Most of these words do not exist in either of the lexicons[30]. From this observation it seemed to us that dialectal words are more likely to appear in online translation than in MSA lexicons. We detail the online translation in the translation step.

[29] http://www.livingarabic.com/
[30] Although inflectional forms of these words might exist, these specific forms were missing.

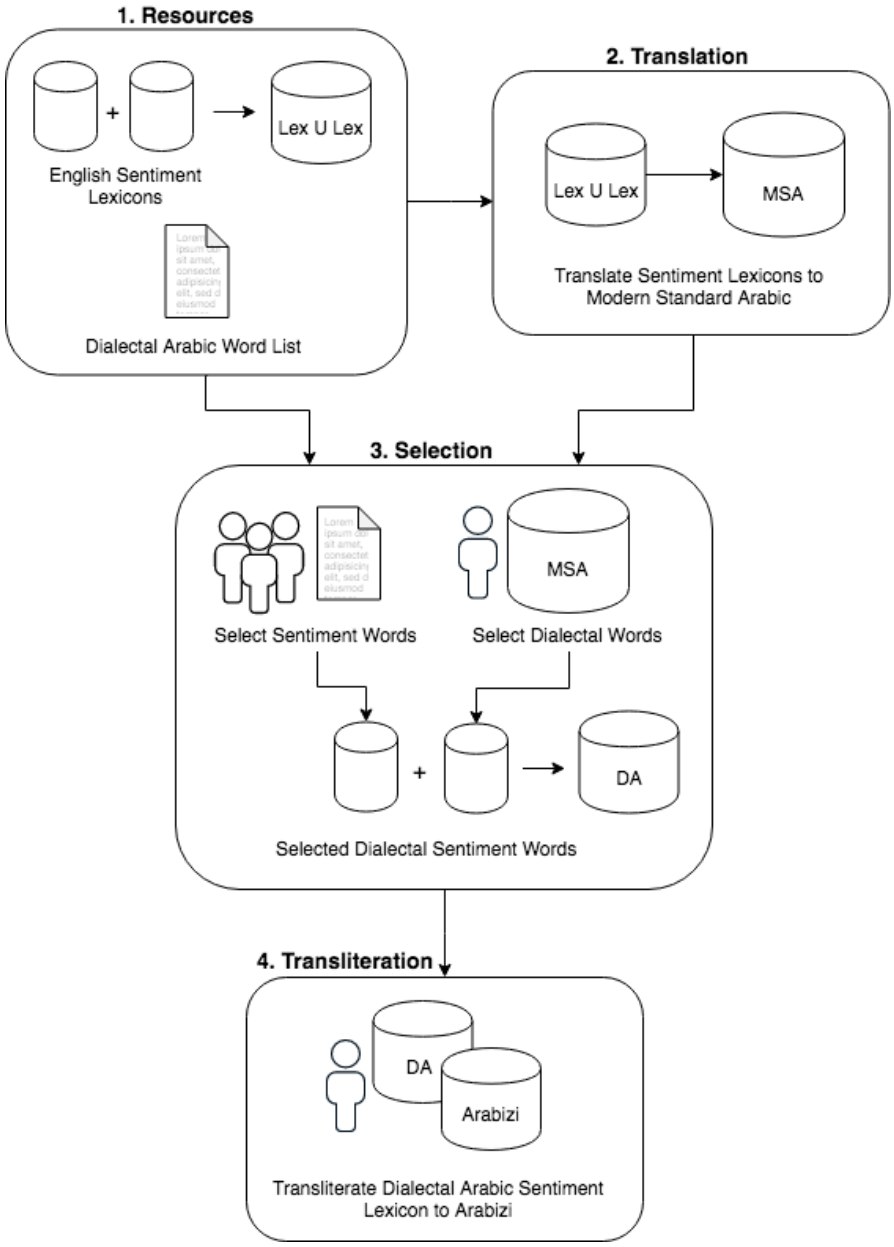


Figure 5.1: Pipeline for creating SenZi

We created a union of the Hiu and Liu and the MPQA sentiment lexicons to encompass all words that are in at least of the two lexicons. For example:

Hiu and Liu:	*joyful, cheerful, excited, inspired*
MPQA:	*cheerful, excited, exhilarated, happy*
Hiu and Liu ∪ MPQA:	*joyful, cheerful, excited, inspired, exhilarated, happy*

We name the combined lexicon HL-MPQA, it consists of 7.8K sentiment words (2.7K positive and 5.1K negative). HL-MPQA is now in position for the next step (Translation). Living Arabic word list will be used in the third step (Selection).

Word from Online Translation	English Meaning	ArSenL	SLSA
مسطول	Idiot	✘	✔
عبيط	Foolish	✘	✔
أهبل	Stupid	✘	✔
متمرد	Insurgent	✘	✘
بطال	A Useless person	✘	✔
متذمر	A negative person	✘	✘
تعجرف	Arrogance	✘	✔
شرس	Fierce	✔	✘
منفتح	Open minded	✘	✘
مبتكر	Innovative	✘	✘
متشائم	Pessimistic	✘	✘
حقير	Despicable	✘	✘
مترفع	Arrogant	✘	✔
امتياز	Excellence	✘	✔
مقرف	Disgusting	✔	✘
مناضل	Not giving up	✘	✘
مضروب	Broken	✔	✘
تفاهة	Silliness	✘	✔
موهوب	Talented	✘	✔
مزعوم	Falsely claimed	✘	✘
فزع	Fear	✘	✔
داهية	Sly	✔	✘
طموح	Ambitious	✘	✔
اعجاب	Adoration	✘	✔
مضياف	Hospitable	✘	✘
طاهرة	Virtuous	✘	✘
اسطوري	Legendary	✘	✔

Table 5.2: A comparison between online translation and MSA lexicons

5.1.3 Translation

Online translation provides a list of synonyms for every input word. With the current scarcity of Lebanese dialect sentiment lexicons, we planned to generate a list of Arabic sentiment words and exploit this list to find the words that are used in the Lebanese dialect. We provide

examples from three online translators: *almaany.com*[31], *bab.la*[32], and *google translate*. We translate the words *success* and *failure* in each of these translators and present them in Figure 5.2.

We observed that all three translators gave similar sets of accurate translations. We were able to find Lebanese words as well among the three translations. We present the Lebanese words from the translators for this example in Table 5.3.

Although all three translators gave words that are common to the Lebanese dialect, one feature makes *bab.la* stand out, that is it provides a phonetic alphabet along with every single-word translation. Figure 5.3 shows the phonetic alphabet for the previous terms *success* and *failure*.

	almaany.com	*bab.la*	*Google Translate*
Success	نجاح	انجاز نجاح توفيق	نجاح توفيق
Failure	تعطل فشل توقف	فشل تعطل توقف	فشل عجز
Number of words	4	6	4

Table 5.3 Examples of Lebanese words from online translation

The automatic (computerised) transliteration has been used quiet often in presenting Arabic in English scientific papers or to process Arabic such as the mentioned ArSenl lexicon (Badaro, et al., 2014). It is an easy replacement of Arabic script with Latin script that includes special characters such as the Buckwalter transliteration system[33]. It uses a distinct representation for the guttural phonemes and heavy consonants such as using upper and lower cases of the same Latin script letter to distinguish a heavy from a light consonant in Arabic. We present some of the Buckwalter representations of guttural phonemes in Table 5.4 and heavy consonants in Table 5.5.

[31] https://www.almaany.com
[32] https://bab.la
[33] https://en.wikipedia.org/wiki/Buckwalter_transliteration

Almaany.com

success (noun): successfulness

فَلاَح ؛ فَوْز ؛ نَجَاح ؛ نُجْح

failure (noun): lack of success

إخْفاق ؛ إفْلاس ؛ تَخَاذُل ؛ تَعَطّل ؛ تَوَقّف ؛ حُبُوط ؛ خَيْبَة ؛ رُسُوب ؛ راسِب ؛ راسِبٌ في ؛ سُقُوطٌ في ؛ ساقِط ؛ ساقِطٌ في ؛ ضَعْف ؛ عُطْل ؛ فَشَل ؛ كَبْو
المختصر

bab.la

success {noun} — إنْجاز · فَلاح · تَوْفيق · نَجاح · شَخْص ناجِح · شَيْء ناجِح

success — كانَ ناجِحاً

failure {noun} — آفة · إفْلاس · اخْتِلال · حُبوط · خَيْبة · خِذْلان · رُسوب · إخْفاق · فَشَل · تَوَقُّف · تَعَطُّل · شَيْء فاشِل · قُصور

failure — شَخْص فاشِل · قُصور في الأَداءِ

Google Translate

Translations of success

Noun

نجاح	success, hit, prosperity, god speed, go
فوز	victory, winning, success, gaining
توفيق	success, accommodation, luck, god speed
ظفر	nail, fingernail, victory, unguis, success
عمل ناجح	success
ألقى نجاحا	success

Translations of failure

Noun

فشل	failure, fail, defeat, fiasco, dud, washout
إخفاق	failure, washout, setback, deadlock, miscarriage, bust
قصور	failure, insufficiency, inability
عجز	inability, disability, failure, deficiency, shortage, incompetence
ضعف	weakness, impairment, fragility, frailty, failure, feebleness
إفلاس	bankruptcy, insolvency, failure, bust, going into liquidation, fall
خيبة	failure, fiasco, discomfiture
تعب	fatigue, tiredness, weariness, lassitude, failure, languor
شخص مخفق	flop, failure
تخلف عن القيام بكذا	failure

Figure 5.2: Examples of online translations of *success* and *failure*

Figure 5.3: Examples of bab.la translations showing phonetic alphabet

Arabic Letter	Phoneme	Buckwalter	Phonetic Description
ح	Ḥā'	H	Voiceless pharyngeal constricted fricative
خ	Khā'	x	Voiceless velar fricative
ع	ʿayn	E	Voiced pharyngeal fricative
غ	Ghayn	g	Voiced velar fricative
ق	Qāf	q	Voiced uvular plosive
ء	ʻ	ʻ	Voiceless glottal stop

Table 5.4: Buckwalter representation for guttural phonemes.

Arabic Letter	Phoneme	Buckwalter	Phonetic Description
ط	Ṭā'	T	Emphatic voiceless dental plosive
ض	Ḍād	D	Emphatic voice alveolar plosive
ص	Ṣād	S	Emphatic voiceless alveolar fricative
ظ	Ẓā'	Z	Emphatic voiced alveolar fricative
ق	Qāf	q	Voiced uvular plosive

Table 5.5: Buckwalter representation for heavy consonant

In Lebanese Arabizi, guttural phonemes are represented as numeric characters or compound letters and there is no distinguishing among light and heavy consonants. For example, as shown in Chapter 2, the خ is *kh* or 5 and both ت and ط light consonant *t* and heavy consonant *t* is *t* without distinction. However, in the Buckwalter transliteration system there is no phonetic distinction among letters that are pronounced differently at different positions in words, because it is a direct mapping of Arabic script with Latin script.

The phonetic alphabet system that *bab.la* uses is the DIN[34] (Deutsches Institut für Normung), German Institute for Standardisation, where it maps Arabic phonemes, the way letters are pronounced, with Latin script. For example, the و *wa* in وسيم *wasīm - handsome* is written as pronounced *wa*, whereas the same letter و is pronounced as *ou* in مجنون *majnūn – insane.* The special character *ū* denotes a long vowel *o* or *ou*. The Buckwalter system does not differentiate between different phonemes of the same letter. For example, since the letter و is mapped with *w*, both of the mentioned words would be transliterated using the same letter *w* where each is pronounced differently: وسيم *wsym* and مجنون *mjnwn*. We present two examples in Figure 5.4 of bab.la transliteration where one vowel appears twice in a word but pronounced differently at different positions.

yāsamīn ياسمين: ي *yā* in the beginning, *ī* at the end

jasmine (also: Yasmin) ياسَمين [yāsamīn] {noun}

wadūd ودود / *friendly*: و *wa* in the beginning, *ū* at the end.

friendly (also: genial, inviting, lovely, warm-hearted) وَدود [wadūd] {adj.}

Figure 5.4: bab.la examples of DIN phonetic transcription in *bab.la*

This phonetic transcription of Arabic in Latin script is very similar to the way Arabizi is transcribed. As such, the DIN transcription of Arabic can be normalised to Arabizi automatically without ambiguating letters such as: *yāsamīn* ® *yasamin* and *wadūd* ® *wadoud*.

[34] https://en.wikipedia.org/wiki/DIN_31635

Since we are resourcing Arabizi for NLP as part of this research course, we chose *bab.la* to create SenZi for the added value of generating a dataset of Arabic words and their phonetic transcription that could be normalised to Arabizi later on. This dataset could be used as a translation matrix between Arabic and Arabizi for a supervised cross-lingual word embeddings (Glavas, et al., 2019), (Vulić & Moens, 2015), (Ruder, et al., 2017) or for evaluating a Levantine Arabizi to Arabic transliteration efforts. Another feature of this dataset is the existence of diacritics on the Arabic script words. This feature is not available in *Google Translate* as shown above in Figure 5.3.

We translated HL-MPQA to Arabic automatically. We wrote a script that fetches every word from HL-MPQA (7.8K sentiment words) and inputs it into *bab.la*. It then extracts the translations (skipping multi-word translations) along with their respective DIN phonetic transcription. For example:

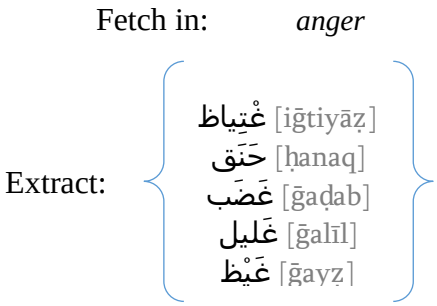


Figure 5.5: Automatic extraction of online translations

We extracted a total 19.2K words (8.4K positive and 10.8K negative), however several input words gave same translations. We therefore filtered out the repeated ones keeping one copy of each word. This reduced the list down to 9.4K words (4.2K positive and 5.2K negative). We added this list containing the DIN transcription to the outcome resources of this thesis. We kept only the Arabic translations to proceed with SenZi. We named it HL-MPQA-Ar.

As mentioned earlier dialectal Arabic is a spoken form of Arabic, esoteric to different Arab regions, hence lacking a standard orthography. The Modern Standard Arabic is the parent to the Arabic Dialects, where they originally derive from. Different words in different dialects goes back to the choice of MSA words within the dialect. For example, among the extracted

translations of *anger*, only one word غضب *ghadab* is common to the Lebanese dialect. Other translations could be common to other dialects. Therefore, we needed to select the Lebanese dialect sentiment words from HL-MPQA-Ar to create a Lebanese Arabizi sentiment lexicon. We also wanted to select the sentiment words from the Lebanese dialect word list, LivingArabic, described in Section 5.1.2.

At this stage we had HL-MPQA-Ar (9.4K translated sentiment words) and LivingArabic (7.1K Lebanese words) in position for the next step (Selection).

5.1.4 Selection

We had two generated Arabic word lists at this step, a union of Hu and Liu and MPQA lexicons translated online to Arabic (HL-MPQA-Ar) and a word list of Lebanese dialect (LivingArabic). We deployed two manual selection tasks based on the human resources that we had:

1. Select dialectal words from HL-MPQA-Ar (9.4K words).
2. Select sentiment words from LivingArabic (7.1K words).

We use the phrase *dialectal words* in the first selection task to refer to words that are common to the Lebanese dialect, as shown in the examples in the previous step.

After selecting the dialectal words from the sentiment wordlist HL-MPQA-Ar and the sentiment words from the Lebanese Arabic word list, we combine the output words from both tasks into one lexicon.

Before we delve into the selection tasks, we talk about Arabic dialectology briefly to show how words are born in Arabic dialects.

Dialect, a particular form of a language which is peculiar to a specific region or social group, as defined by Google.

In Chapter 2, we showed the different varieties of spoken Arabic. It is branched into around 20 major dialects. Some of these dialects are influenced by foreign languages. Maghrebi

dialects[35] for example are influenced by French, Spanish, and Amazigh, the language of the Berber. The Levantine and Egyptian dialects are influenced by Turkish. Hence, words in several Arabic dialects could be taken or derived from MSA or borrowed from an influencing foreign Language. The modern Lebanese Arabic is influenced by Turkish, French, and English due to the Archaic Ottoman and French ruling in the region and the modern American westernization of education and media. We provide examples of Lebanese greetings, positive, and negative words categorised by the Language taken from:

Modern Standard Arabic:

1. Taken as is: نجاح *success* and فشل *failure*

2. Derived: مسطول *idiot* from سطل *bucket* (as useless as an empty bucket)
   جولق *mock* (مجلوق, جلجقة) forms of a *spoiled character* from جلق *laughter*
   استذة *perfectly done* from استاذ teacher or expert

English:

mdaprass from *depressed*
mpannak from *panic*
m2angar from *angry*
luvvik from *love-you*
missik from *miss-you*

French:

Bonjour, bonsoir, bonnuit, salut, merci, cava
Good morning, good evening, good night, hello, thanks, good

Turkish:

اسطة *expert* from üsta
خواجة *elegant* from *hoça*
بلطجي *gangtser, unjust* from *baltaci*
قاووش *mess* from *kavoş*

As such selecting words from a list of Lebanese dialect words and a list of online Arabic translation to create a sentiment lexicon increases the chance to posses both types of vocabulary, words of MSA and of foreign origins.

[35] https://en.wikipedia.org/wiki/Maghrebi_Arabic

5.1.4.1 Select dialectal words from HL-MPQA-Ar

Identifying Lebanese words automatically requires Lebanese datasets to either train a Lebanese dialect language model or simply search each word in a Lebanese dialect corpus. As such, the current lack of Lebanese Arabic lexical resources, the formation of SenZi necessitates a word selection task. To the best of our knowledge that was the only way to correctly identify which words are common to the Lebanese dialect among other Arabic translations. Additionaly, handcrafting the lexicon by Lebanese natives produces a more reliable lexical resource, hence a compensation of time for quality.

In any case, as shown in the previous example of translating the word *anger* (Figure 5.5) on *bab.la* gives a set of words of which one of them غضب is common to the Lebanese dialect. As such selecting the Lebanese words is a fairly simple task, because it is very unusual for the rest of the words to appear in the Lebanese dialect. Given the simplicity of this task and the limited human resources, we assigned this task to one Lebanese native volunteer student.

We provided the student with the list of HL-MPQA-Ar and asked them to select the Lebanese dialect words. Out of 9.4K words (4.2K positive and 5.2K negative), the student selected 537 words from the positive (13%) and 1K words from the negative lists (19%).

This makes up the first portion of the Lebanese sentiment lexicon, we now detail the second selection task.

5.1.4.2 Select sentiment words from LivingArabic

LivingArabic is a list of Lebanese Arabic words developed within the LivingArabic project that does not contain polarity scores or sentiment labels. The aim of this step is to exploit this list to find Lebanese dialect sentiment words to build SenZi.

However, since the decision whether a word is positive, negative, or neutral could be subjective to the decision maker and depends on the context of the words in the sentence, we decided to be more careful than the previous selection task. Similar to the creation of Twitter

datasets (Chapter 4), we assigned this task to three student volunteers. The motivation of having three students select the sentiment words is to increase the chance of making a more accurate selection by generating several sentiment opinions for every word and selecting the ones that the majority agree upon. For example:

The word كبر out of context could either be referring to a negative attribute of *looking down onto people* (*arrogance*) or neutral *grow in age or size.* If the first and second students consider different meanings for the word, then the third student breaks the tie. Hence, every word receives three opinions and we select what two annotators agree upon, negative in this example. We present this in Figure 5.6.

We presented the LivingArabic list of 7.1K words to each of the three students. We asked them to go through the list word by word to check whether each word imply a sentiment, if so, label the word with *P* or *N* (short for *positive* or *negative*), otherwise if *neutral* or *ambiguous*, skip the word.

Out of 7.1K words, the three students selected 533, 672, and 1033 sentiment words each. We took the words that at least two students agreed on their polarity for SenZi. That is 179 positive (4.3%) and 553 negative (10.6%). This makes up the second portion of the Lebanese sentiment lexicon. We present this selection in Table 5.6.

	Student1	Student2	Student3	2-Student Agreement
Sentiment Words	533	672	1033	732
Positive	155	177	268	179
Negative	378	495	765	553

Table 5.6: Dialectal Words Selection

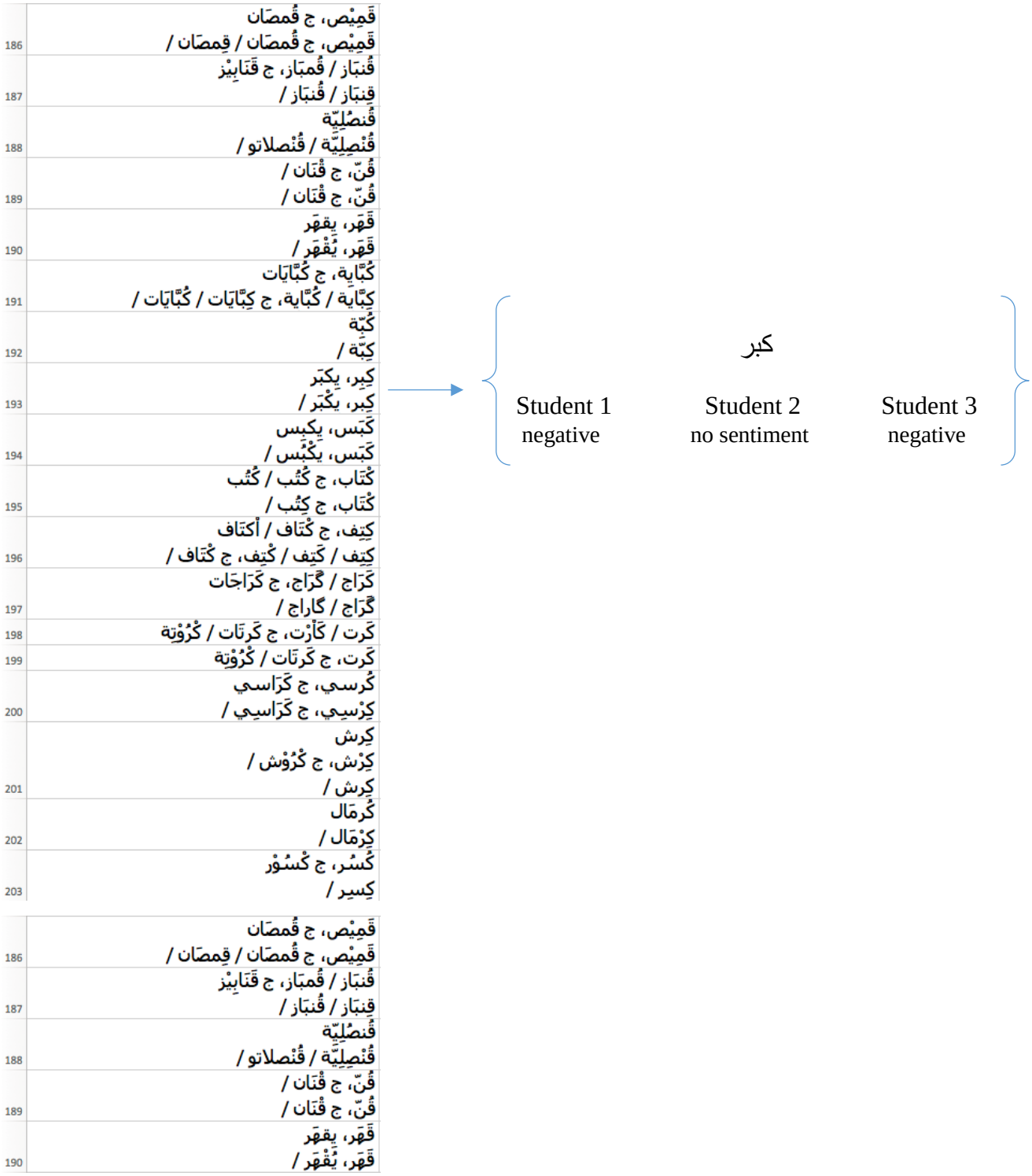


Figure 5.6: Selecting sentiment words from LivingArabic word list.

As a result, the selection tasks produced two Lebanese Arabic sentiment lists:

1. 1.5K dialectal words (16.3%) from HL-MPQA-Ar (9.K words).
2. 732 sentiment words (10.3%) from LivingArabic (7.1K).

Similar to the fusion of the English sentiment lexicons Hu and Liu and MPQA in Section 5.1.2, we take the union of these lists. The union resulted in a Lebanese sentiment lexicon of around 2K words (607 positive and 1.4K negative).

No further investigation on how the Lebanese Arabic script lexicon performs in sentiment analysis because of the lack of public sentiment-annotated Lebanese data during the time of developing this lexicon. However, we add it to the list of outcome resources from this research.

This is the first version of the sentiment lexicon, but it is in Arabic script. In the next and final step (Transliteration), we transliterate it, as is, to the Latin script Arabizi.

5.1.5 Transliteration

Dialectal Arabic is a spoken language, hence there is no consistent orthography in transcribing it in Latin script, a major challenge for the sentiment analysis of Arabizi, discussed in detail in Chapter 2.

The way Arabic is Latinised in Arabizi could not be encapsulated in a set of letter to letter mappings from Arabizi to Arabic script or vice versa. One major factor to this limitation is the inconsistent occurrence of vowel letters in Arabizi, because there are short and long vowels in the Arabic script where short vowels are not letters but diacritics, diacritics that are usually not written in social Arabic as well. For example:

جَميل pretty - jamil

The diacritic above the first letter ج *ja* is the short vowel *a*. This word would be written جميل in the social text without the diacritic, therefore a rule-based transliteration would give *jmil*, very unusual to the Lebanese dialect Arabizi.

Another factor is that most of the time, the way Arabizi is transcribed reflects the way it sounds (phonemes), not the way Arabic script looks like (graphemes). As shown in the

translation step in Section 5.1.3, the phoneme of a single Arabic letter differs in different positions in the word. For example:

The ي is closer phonetically to the Latin *i in* جميل *jamil*, but closer to a *ya* in يسير *yasir* - *facilitated.* Hence fixing mapping rules would generate an error in either of the cases: *jmyl / isir* both transliterations lost the syntactic, semantic, and phonemic structure of the word.

Since a rule-based automatic transliteration does not present the words the way they are written naturally. We hand-transliterated every word to Arabizi.

If the Arabizi orthography is inconsistent, that every word could be written in different ways, *how can a list of Arabizi words that contains one orthographic form for each word match the wide forms of these words in social media data?*

As mentioned earlier, we planned to start with one natural (not computer-generated) form for every sentiment word and then try to retrieve the orthographic forms automatically using word-embeddings in the next stage of SenZi, expansion. For example, all of the following are common orthographies for the word خير *kheir* - *fine* or *good*:

Kher, kheir, khayr, khyr, khair, kheer, 5ayr, 5eir, 5air, 5er, and *5yr*.

Since they are all syntactically and semantically related and often used in text, their vector representations should be close to each other in the embedding space. Based on this information and the size of our Facebook corpus (1M Arabizi comment), we assumed that having any one of these forms in SenZi should not be a major issue, since we plan to retrieve the rest of the forms in the expansion.

As such, having three students transliterate the words is very unlikely to add lot of value. We wrote the positive word *kheir* in 11 spellings. There is no correct or wrong way of spelling Arabizi, it is a policy-free language.

We assigned the transliteration of the dialectal sentiment lexicon to one Lebanese native who uses Arabizi regularly. Noting that different mappings of Arabic phonemes in Latin script may be used interchangeably such as the خ in the mentioned word خير could be mapped with

compound letter *kh* or numeral 5 as in *kheir* and *5eir*. We asked the student to transliterate the lexicon word by word the way they naturally transcribe Arabizi, without setting any orthographic instructions. The student transliterated the Lebanese sentiment lexicon consisting of 2K words (607 positive and 1.4K negative) to Arabizi marking the birth of SenZi.

In the next stage in Chapter 6 we expand SenZi by retrieving the orthographic and inflectional forms of the generated sentiment words. Finally, in Chapter 7 we present the sentiment analysis evaluation of both, SenZi and its expanded versions.

5.2 Discussion

In this chapter we detailed the creation of a new resource for a Levantine Arabic dialect, SenZi, a sentiment analysis lexicon for Arabizi. We now highlight some limitations along the development of the lexicon.

The lack of sufficient annotated corpora and datasets for the Lebanese dialect necessitated a few manual selection tasks in search for Lebanese sentiment words. Manual selection is costly in terms of time and availability of human resources, hence a drawback for replication, however, at the expense of producing a more accurate and reliable language resource over the automatic selection or transliteration.

In the selection task of Section 5.1.4.1, selecting dialectal words from HL-MPQA-Ar, we had one student volunteer to carry out this task, therefore the outcome is biased to the student's opinion and linguistic cognition, however, the student selected 1.5K dialectal words out of 7.1K (16.3%) of the HL-MPQA-Ar. Though one volunteer might have missed some dialectal words, we consider resourcing a low-resourced language with 1.5K sentiment related words a good initiative for building more resources later on to fill the NLP gaps.

In the selection task of Section 5.1.4.2, selecting sentiment words from LivingArabic, we had three student volunteers selecting the sentiment words from LivingArabic (7.1K words) list. Table 5.6 showed that the selection results varied among the three students. A direct effect of

classifying sentiment of words out of language context. We checked the disagreement to learn that these words possess contextual meanings, in some cases inferring positive or negative connotation but neutral in other, depending on their contexts in the text. For example:

2aber قبر:	grave or beat harshly
shak شك:	dive in or doubt
sa77a صحة:	cheers or health
kalb كلب:	dog or an insulting expression

The selection depended on how the students perceived such words. Identifying words out of textual contexts as positive or negative could be very inaccurate given the words' polysemic nature. An alternative, possibly more accurate, approach is to, for every word retrieve a number of short sentences such as tweets containing the word, ask the students to record a sentiment score to these sentences, and average the results to score the word in an attempt to capture the impact it has on the sentences. However, under the limited resources and the volunteering time, we took the words that the majority agreed upon.

In the transliteration task in Section 5.1.4, we had one student volunteer to transliterate 2K Arabic words. We presented our analysis to prove that an automatic letter to letter mapping transliteration fails in the case of Arabizi for its natural orthography, more on this in Chapter 2. Another possible approach to the automatic transliteration of different scripts is sequence-to-sequence generation. A neural network that saves information such as LSTM can be trained on parallel data, words of both scripts, for some time, and predict a transliteration for new words (Rosca & Breuel, 2016). Reverse transliteration in this case from Arabic to Arabizi might be less ambiguous than Arabizi to Arabic because each light and heavy consonant letter would map to a single Latin script letter, however, this training requires a parallel dataset. The outcome resource of this task could be used for future automatic transliteration efforts.

Nevertheless, Lebanese natives invested their time in handcrafting SenZi to produce a reliable resource for the sentiment analysis of Lebanese Arabizi.

5.3 Chapter Summary

In this chapter we tackled RQ2 by presenting SenZi, a new sentiment lexicon for the Lebanese dialect Arabizi. We used some lexical resources from the literature to create SenZi through several stages of manual and automatic steps that consist of translation, transliteration, and selection.

The resulting lexicon contains 2K sentiment words, around 600 positive and 1.4K negative. Given the high degree of sparsity in Arabizi, we consider this lexicon to be relatively small to match the large magnitude of inflectional and orthographic forms that each sentiment word may have. As such, we present automatic expansion techniques for SenZi in Chapter 6 to cover a large number of forms for each sentiment word present in SenZi.

We fully address RQ2 in Chapter 7 by using SenZi in a lexicon-based classification approach to evaluate its value for the Arabizi sentiment analysis.

6 Lexicon Expansion

يا مسكّنى وسكنى وسكينتى وساكنتى وسكونى وسكوتى

وسكّتى وسكرتى وسكّرتى وسرّى وسريرتى وسروري وسروري

فمن لروحى وراحى يا أكثرى وأقلّى

ويا كلّ كلّى فكن لى ان لم تكن لى فمن لى

Arabizi, the Latinised Arabic that inherits Arabic's language structure, as to Arabic it is morphologically rich but unlike Arabic it orthographically rich as well.

In Chapter 2 we showed the layers of Arabic morphology where lemmas derive from triliteral stems and inflections derive from lemmas or from stems directly. Let alone the sparsity caused by the nature of the language, this sparsity is multiplied by the inconsistent orthography of Arabizi. Every lemma and every inflection has a range of possible spellings. For example:

محبوب *ma7boub - beloved,* a lemma of حبّ *7ob - love*.

ma7boub: mahboub, mahboob, ma7bub, ma7bb, mahbub, m7boub, mhboub, m7bub, mhbub, mhboob, m7boob, ma7boob, m7bob, mhbob, mhbb etc..

And since Arabizi is a social text, like other languages on social media, each orthographic variant may be exaggerated as well such as:

ma7boub: m7booouuubbb, m7boubbbiiiiii, etc..

Since sentiment analysis aims to identify text as positive or negative from the word structure of the text, the high degree of sparsity in Arabic pose a major challenge on sentiment analysis.

SenZi contains 2K sentiment words written in a single orthography. In its current structure it is incapable of capturing the wide range of inflectional and orthographic forms of its sentiment words. We propose to expand SenZi by enriching it with inflectional and orthographic forms automatically using word-embeddings to start addressing RQ3.

Could word-embeddings enhance the performance of Arabizi sentiment analysis?

Expanding SenZi means finding as many forms as possible for each sentiment word and adding it to the lexicon. In this chapter we introduce word embeddings and explain how we propose to use this deep learning technique to retrieve the inflectional and orthographic forms of the sentiment words in SenZi. We also added to this process a new word matching approach that filtered in the most relevant words to SenZi.

Using the word embeddings along with the matching approach together and separately in different configurations, we created six new expanded versions of SenZi. We detail each expansion in this chapter. We finally evaluate SenZi and its expanded versions in Chapter 7 using a lexicon-based sentiment analysis approach.

Before delving into the details of the word embeddings approach, we answer the following question: *Are there other approaches to address the lexical sparsity?*

We list two approaches that we overlooked for their limitations:

1. Regular Expressions
2. Stemming

Regular expressions (regex) is a powerful text manipulation method for editing and searching. It consists of a set of metacharacters injected within words to automate the matching process of specific patterns in text. For example, the word *7abibi - my-love* could be written using regex to match the different spellings in a regex search. We dissect the meta-characters below:

7abibi: [7h]a?b+([ie]+)?b+([ie]+)?

[7h]	Match a 7 or an *h*
a?	Match with or without the *a*
b+	Match one *b* or more
([ie]+)?	Match with or without *i* or *e*, or combination of both letters, even if repeated

This regex sequence matches all of the following variants of the word *7abibi*:

7bibi, 7abibi, 7abibiiiii, 7bb, 7bb 7abbiieeebbbbiiiii 7abebe 7abeb 7abbbbbeeebbb, 7abiebi, habibi, hbbbb, habibiiiii, etc..

The major limitation to this approach is that regex has to be hard coded into every sentiment word in the lexicon because each word has different inflectional and orthographic patterns at different positions in the word. As such building a lexicon of regex is costly to maintain and update thus inefficient.

The other limitation is that although adding a sequence of meta-characters into words enables matching forms of these words, it risks matching irrelevant words that contain same character combinations.

Instead of retrieving several forms for every sentiment word, *why not stem every form, so it could be matched with the root of the word?*

First, the stem of the word does not necessarily indicate the same sentiment of that word. Lemmas and stems could have opposite sentiment. We present some examples below from the Lebanese dialect:

شيخ: متمشيخ	رجل: مرجلة	شاطر: شطارة	قتل: قتال مقاتل
religious preacher: deceiver	man: unjust or arrogant	clever: slyness	kill: fight, fighter

Second, as shown in Chapter 2, Arabic inflections are not limited to a set of prefixes and suffixes. Apart from the proclitics and enclitics, a word could be inflected by a combination of affixes and diacritics including infixes as well. The large vocabulary of triliteral stem

words along with the rich morphology in Arabic makes it difficult to extract the correct stem from inflections. We give examples below with a letter to letter transliteraion for clarity:

Act of denying: استنكار

Trilateral combinations: استنار انار ستر ستار سار سكر كر تنك تنكر نكر نار
eleven words of different meanings

I/he-supports: بشجع bshj3

Trilateral combinations: بشع bsh3 - *ugly* or شجع shj3 - *encourage*

Third, a root letter may be dropped in an inflection, for example:

Hunger: جوع jou3.

We got hungry: جعنا j3na, the root letter و dropped. (Lebanese Dialect)

Up to our knowledge there are no known public computational stemmers with high stemming accuracy. Other stemming efforts are dictionary based for MSA not dialectal Arabic such (Smrž, 2007), (Pasha, et al., 2014).

Nevertheless, we propose to address the high degree of sparsity in Arabizi to cover the inflectional and orthographic forms of the SenZi sentiment words by expanding SenZi automatically using word embeddings.

6.1 Word Embeddings

Word embeddings is the name given to a deep learning architecture consisting of neural networks that embeds words into vectors of real numbers projected in a vector space. It received great attention in NLP for its powerful applications since the release of word2vec by Google dominating the state of the art (Mikolov, et al., 2013). It has been used in recommendation systems, language models, clustering, topic discovery, and translation.

A neural network embeds the vocabulary of an unsupervised corpus into vectors consisting of features, also known as parameters, about the words in real numbers. Features such as the

relationship of every word with the rest of the words in the corpus. Embedding words into vectors in a vector space sorts the words according to their meanings. Vectors of words that co-occur frequently are projected near each other in the space. As a result, similar vectors would be clustered together such as movie names, countries, greetings, fruits, smartphones, political terms, etc. Not only words within the same vector clusters are related in meaning but also the distance separating the vectors indicate a relationship between words, for example the distance between the vectors UK and London might be equal to the distance between China and Beijing in a given corpus.

Finding similar words through their embeddings leveraged language models for NLP tasks such as word prediction used in emails and mobile messaging. A language model trained on a corpus predicts the *next word* by generating a vector of that corpus vocabulary and ranks each word according to its vector similarity with the neighbouring words of the *next word* and the number of times these words co-occurred together in the corpus. For example:

The quick brown fox jumps over the lazy …

…	*…*
rodent	*0.01*
otter	*0.01*
dog	*0.89*
duck	*0.2*
cat	*0.3*
rat	*0.04*
…	*…*

Figure 6.1: Word Completion Example

Creating an embedding space, or training a model on a corpus, can be achieved in a continuous bag of words (CBOW) or a skip-gram fashion:

Given a range sequence of words (context), a CBOW neural network predicts the probability of a word within the context as shown in Figure 6.2.

On the contrary, a skip-gram model predicts a sequence of words, context, within a certain range, given a word from that context as shown in Figure 6.3. Skip-gram represents rare words well and works better than CBOW in less amount of data (Mikolov, et al., 2013).

We planned to exploit the power of word embeddings to find syntactically related words of SenZi in our search for orthographic and inflectional forms. This requires a large corpus of Arabizi conversations. We used the Arabizi Facebook corpus that we created in Chapter 5 for this purpose. We trained word embeddings models on the corpus to create an embedding space. We search the vectors that represent each SenZi word in the space and extract the vectors (words) surrounding it, known as nearest neighbours. We tested this approach to retrieve the inflectional and orthographic forms of the input SenZi words. We show the planned steps in Figure 6.4.

W_{t-2} W_{t-1} W_t W_{t+1} W_{t+2}

Predict W_t using its vector relation with surrounding words.

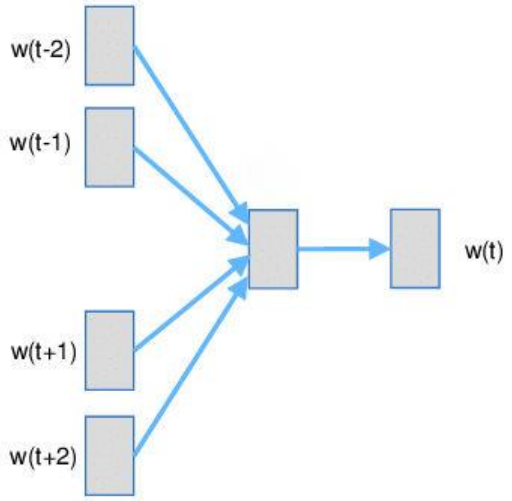


Figure 6.2: CBOW Word Prediction

W_{t-2} W_{t-1} W_t W_{t+1} W_{t+2}

Predict surrounding words of W_t using their vector relation.

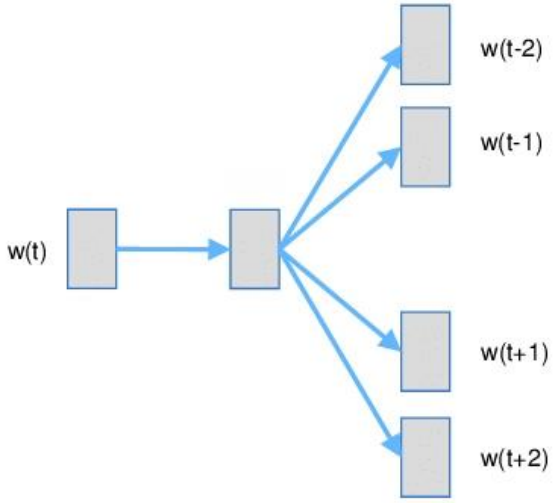


Figure 6.3: Skip-Gram Word Prediction

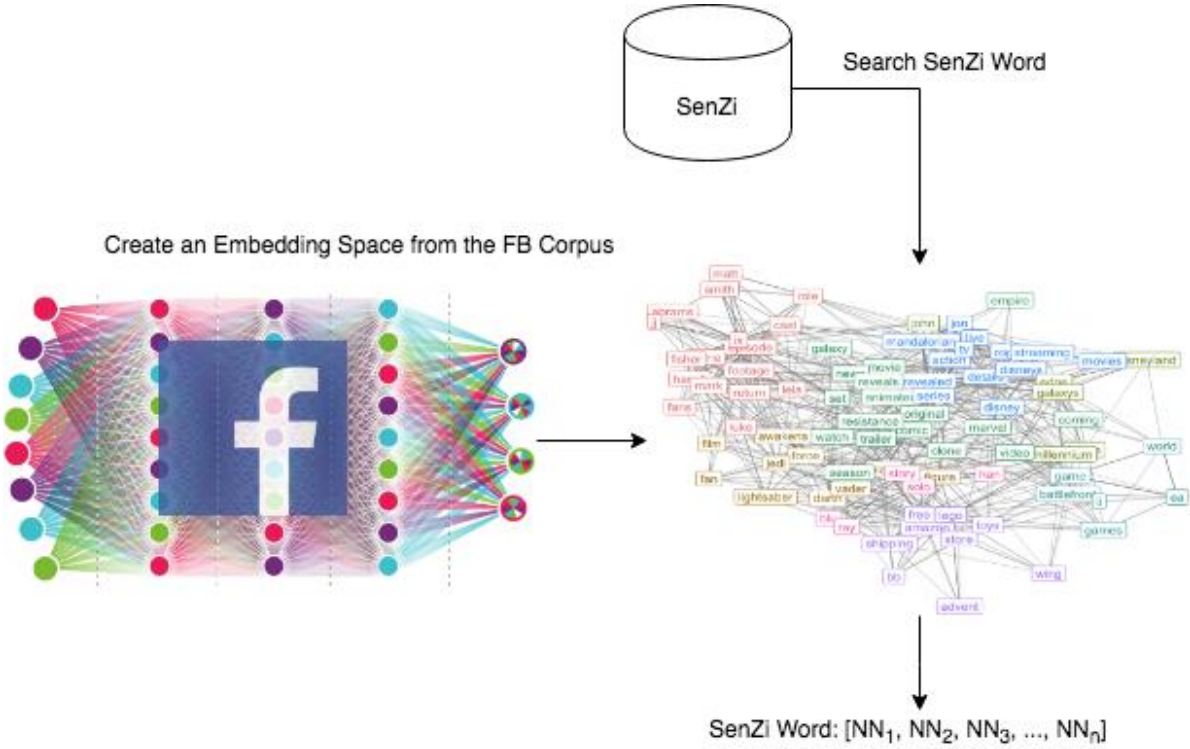


Figure 6.4: Expanding SenZi

6.1.1 Nearest Neighbours

Creating an embedding space from the FB corpus can be done via one of the public word embedding models provided by AI organisations such as Google and Facebook. We start by

using word2vec (Mikolov, et al., 2013) following the approach presented in Figure 6.4. We note that a corpus of 1M comments is considered small to take full advantage of the embeddings (Chapter 3). We however tested it by searching some SenZi words and extracting 20 nearest neighbours.

We searched three SenZi words *7abibi - my-love or darling, ndif* means *clean,* used as *virtuous,* and *wesikh* means *dirty,* used as *rude*. For each of these SenZi Arabizi words, we list their nearest neighbours (NN) then present a translation of the Arabizi word and the nearest neighbours below it. However, for every nearest neighbour that is a form of the SenZi word, we translate it as *word-form* to show that is a morphologic or orthographic form of the SenZi word.

7abibi: 7abibi habibi habibe kbeer m3alem ammi 5aye teslam 7abibe raytak 7bb Hbb a5i 7abib 5ayi kbiir waynak Habibi kbirrr joud

my-love: 7abibi, 7abibi-form, high-rank, leader, uncle, brother, welcome, 7abibi-form, wish-you, 7abibi-form, 7abibi-form, brother, 7abibi-form, brother, high-rank, where-are-you, 7abibi-form, high-rank, name.

ndif: ndif bidal fiyo ha2na fesid ntaha essmo ghayro terikho hayalla fechel ouat howee 3indo a3la tayaro cha2fit yente5ib tehdid byehterim

virtuous: virtuous, remains, within, our-right, corrupt, ended, his-name, other, his-history, whichever, loser, forces, him, with-him, higher, expelled, piece-of, he-votes, threat, he-respects

wesikh: *wesikh elkaleb 7ayawen. wosikh ra2bit jabein khayin charafo kalebe kalb! monhat barbare ghachim nekir kheyen mret 7a2ir 5eyin 7a2oud 5erfen*

rude: rude, the-dog, animal, rude-form, responsibility, coward, traitor, his-honour, dog-feminine, dog, low-profile, barbaric, idiot, denier, traitor, useless, despicable, traitor, envious, insane.

We can see from these examples that there were 6 forms of *7abibi,* none for *virtuous,* and 1 for *rude* from their retrieved nearest neighbours. Majority of the words are related

contextually with the SenZi words but not morphologically or orthographically. This is because the parameters of the embeddings are based on the context of the words, meaning co-occurrence scores or probabilities with the rest of the word vocabulary.

Facebook developed FastText (Bojanowski, et al., 2017), an updated extension of word2vec that focuses on the structure of the words. Apart from the context of words they added subwords to the embeddings parameters. Therefore, words with similar structure will cluster together in the embedding space as well. We present example subwords of the word *7abibi* in Table 6.1.

FastText takes all subwords between the size of 3 and 6. They said that they have modelled the morphology by adding the subword parameters, and that skipgram works better with subword parameters than CBOW. As such we used fastText to create a new embedding space and repeat the same experiment shown in Figure 6.4.

Subwords	Subword Size
7a ab bi ib bi	2
7ab abi bib ibi	3
7abi abib bibi	4
7abib abibi	5

Table 6.1: Subwords of *7abibi*

We searched the same three SenZi words *7abibi* (*my-love or darling), ndif (virtuous, clean*), and *wesikh (rude, traitor, dirty)*. We also present the SenZi words with their corresponding nearest neighours and a translation of their meanings:

7abibi: 7abibo 7abibii 7abibe 7abibit 7abibak Hbibi 7abibeti 7abibiii 7abib 7abibeh 7abibet habibi 7abibt Habibi 7abiba 7abibty 7bibi 7abibna 7abi 7abibete

my-love: 7abibi-form, 7abibi-form, 7abibi-form, …, 7abibi-form.

ndif: ndif! lndif lendif tndif ndif.. ndifi ndiff nadif tendif nedif chemim zarif ndife chemo naddif chemi5 wza3im ndir chemikh za3im.

Virtuous: virtuous-form, clean-form, virtuous-form, …, smelled, cute, virtuous-form, smell, clean-form, glorious, and-leader, we-manage, glorious, leader.

wesikh: lwesikh wosikh wessikh wossikh wasikh wesi5 wessekh wessi5 wesekh wassikh wesse5 wissikh wesekh. wisikh wesik weskh sikh wesi2 wesi3 wosi5

rude: rude-form, rude-form, rude-form, …, confident, sikh, confident, spacious, rude-form.

All twenty nearest neighbours of *7abibi* are forms of *7abibi,* either inflectional or orthographic. There were twelve forms of *ndif,* and sixteen of *wesikh*. We noticed some irrelevant words among the nearest neighbours that had similar word structure as the input SenZi word. For example:

ndif: ndir - we-manage
wesikh: wesi2 - confident, wesi3 - wide

As can be seen the irrelevant words are very similar in their subwords with the SenZi words. One letter difference in these cases. This indicates that the model positions words with typos near their correct forms, a good feature for retrieving correct words from words written with typos. However, syntactically related words with one letter difference, although very few in these examples, could have an opposite sentiment, as in the example, *wesi2 - confident* neighbouring *wesikh - rude*. Hence copying all nearest neighbours blindly harms SenZi.

We needed to copy all relevant (inflectional and orthographic) words automatically while minimizing the error (irrelevant words) into SenZi. In the next section we describe an approach that we learned heuristically to match the desired syntactically related words.

6.1.2 Consonant Letter Sequence Matching

We learned by observation that a neighbouring word is an inflectional or orthographic form of the SenZi word if it contains the same sequence of consonant letters. If we take the consonant letter sequence (CLS) of *wesikh - rude* for example and match it with its nearest neighbours.

Nearest Neighbours of *wesikh: lwesikh wosikh wessikh wossikh wasikh wesi5 wessekh wessi5 wesekh wassikh wesse5 wissikh wesekh. wisikh wesik weskh sikh wesi2 wesi3 wosi5*

As we said from its twenty nearest neighbours, sixteen are relevant and five are irrelevant. We show the nearest neighbours that matches the CLS of *wesikh* (*wskh*).

wesikh (wskh): l<u>we</u><u>s</u><u>ikh</u>, <u>wo</u><u>s</u><u>ikh</u>, <u>wi</u><u>s</u><u>ikh</u>, <u>wa</u><u>s</u><u>ikh</u>, <u>we</u><u>skh</u>, <u>we</u><u>s</u><u>ekh</u>

Regardless what comes before or after the CLS, as long as no consonant letters intervene within the sequence, we consider the word a match. For that, *lwesikh* for example, the proclitic *l* (*l+wesikh)* means *the rude one*, matches *wskh*, because it contains the same CLS.

Using this approach six relevant words out of fifteen matched the mentioned SenZi word. The irrelevant neighbours do not match as they are structured with different consonant letter sequence than *wskh.*
wesikh: wesik sikh wesi2 wesi3

As for the remaining ten relevant words that did not match, shown below, they all contain the same sequence of consonant letters *wskh* but because the Arabizi orthography is inconsistent these consonant letters are transcribed differently.

wesikh: wessikh wossikh wesi5 wessekh wessi5 wassikh wesse5 wissikh wesekh. wosi5

The observed orthographic difference falls in three categories:

1. A different transcription of a consonant letter phoneme. For example, the guttural خ *kh* in وسخ *wskh* is transcribed as compound letter in SenZi but as the numeral 5 in some of the nearest neighbours *wesi5 wessi5 wesse5 wosi5*.

2. A double letter or more to transcribe a gemmination, a diacritic that emphasizes a phoneme, or an exaggeration. For example: وسّخ *wsskh*, a verb form of *wskh - to dirty or ruin, wessikh wossikh wessekh wessi5 wassikh wesse5 wissikh* or *wesikhhhhh!*.

3. Word contains non-alphabet character such as *wesekh.* with the full stop.

We address these issues by adding light normalization steps before matching the SenZi words with their nearest neighbours. We describe the steps below and provide examples.

1. Compound Letter Replacement:

   We replace the compound letters *gh, kh,* and *ch* or *sh* (غ خ ش) with single characters 8, 5, and $ respectively for both, the SenZi word and the nearest neighbours.

   Arabizi users use the letter 7 or *h* to transcribe the pharyngeal ح, however *h* is a transcription for a similar phoneme ه, we therefore replace letter *h* in the nearest neighbour with the more accurate 7 only if the *h* is at the same position as the 7 in the SenZi word.

   These transcriptions are used interchangeably in Lebanese Arabizi based on our study of the transcription detailed in Chapter 2.

   wossikh ® wossi5
   habibi ® 7abibi

2. Repeated Letters Reduction:

   We reduce repeated consonant letters, two or more, to one in the nearest neighbours.

   wesssikh ® wesikh

We then match the normalised nearest neighbours with the normalised SenZi word if the nearest neighbours contain the same CLS of the SenZi word.

SenZi: *wesikh ® wesi5* (*ws5*)
NN: *mwassa5 ® mwasa5* (*mws5*)

We note that after matching the normalised nearest neighbours with the normalised SenZi word we copy the original nearest neighbour words to SenZi. The normalisation layer is just to find the related words. Therefore, we name these steps hidden layer, because the normalisation does not impact the matched words.

Finally, we filter the matched original nearest neighbours from non alpha numeral characters such as punctuations and emojis. We present this approach in Figure 6.5. In the figure we use the word ARABIZI to denote an example of a senzi word, and RBZ for the CLS of ARABIZI. For every normalisation we use NN' to denote a normalised nearest neighbour, NN'' (second normalisation) and so on. $CLS_{NN''}$ means the consonant letter sequence of the normalised nearest neighbour.

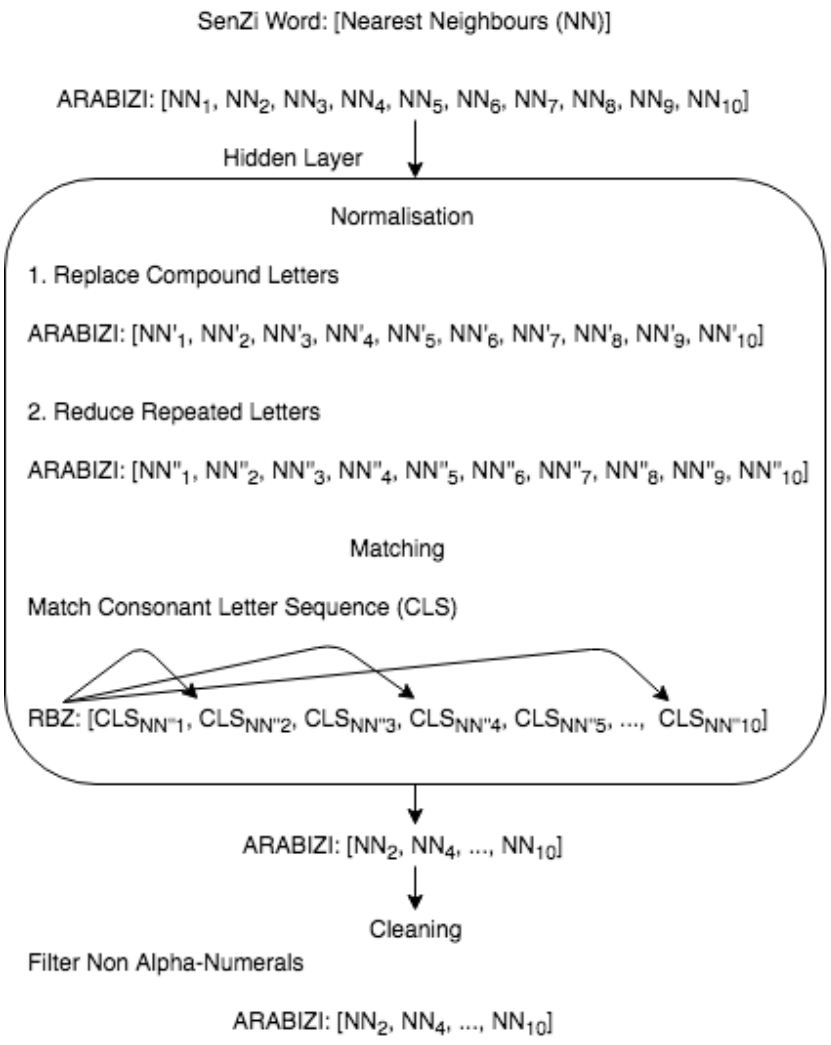


Figure 6.5: Nearest Neighbours Filtering (Hidden Layer)

The CLS matching topped with the mentioned normalisation matched all related words of *wesikh*.

wesikh: lwesikh wosikh wessikh wossikh wasikh wesi5 wessekh wessi5 wesekh wassikh wesse5 wissikh wesekh. wisikh wesik weskh

We add this layer, hidden consonant-letter-sequence (CLS) matching, to the original expansion diagram for clarity, presented in Figure 6.6.

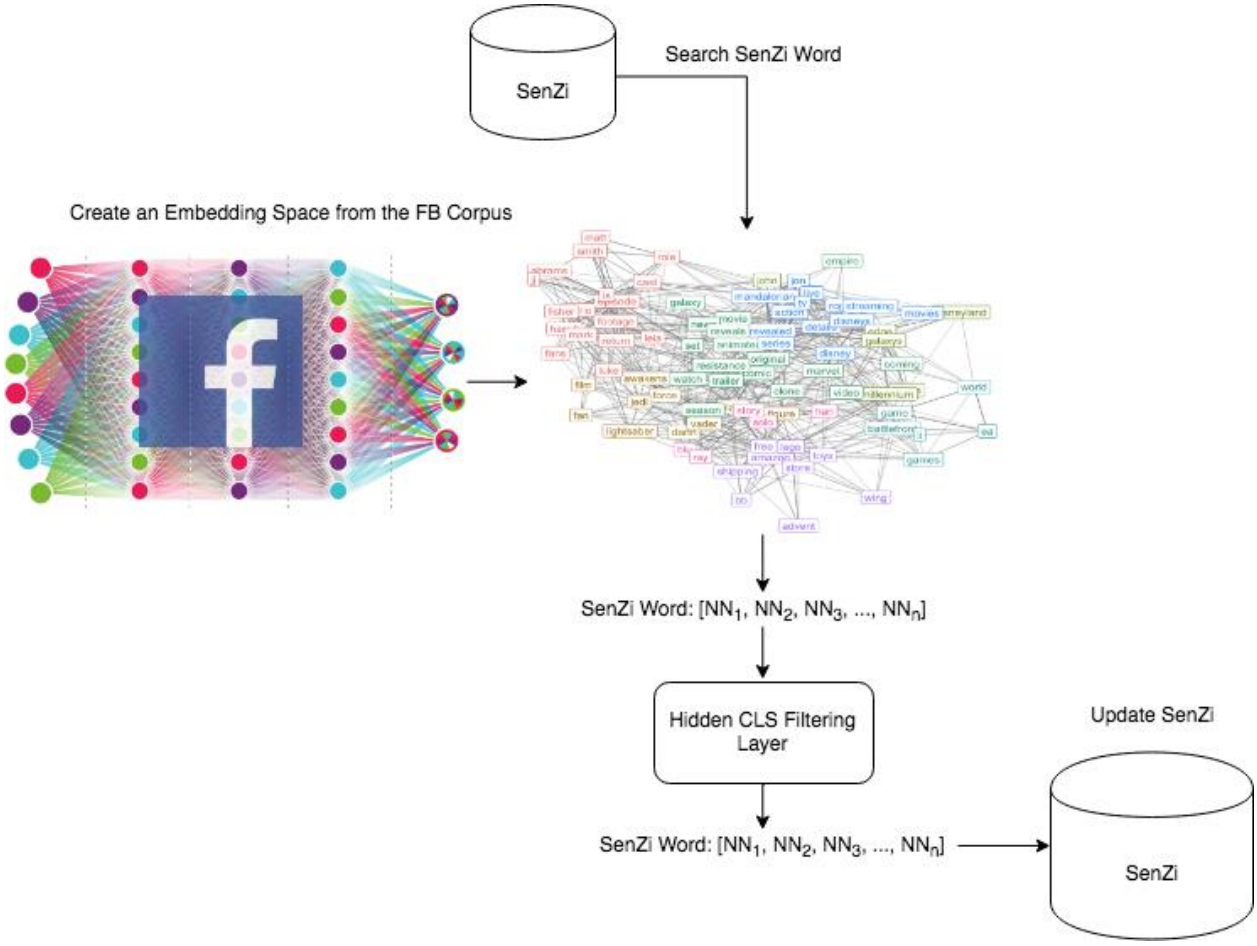


Figure 6.6: SenZi Expansion with CLS Filtering

6.1.3 SenZi Expansions

The designed approach utilises word embeddings combined with the heuristic CLS matching method. The advantage of word embeddings is that it retrieves words that are syntactically related, potentially orthographic and morphological forms. The advantage of CLS matching is to automatically select words that are also potentially related but transcribed differently thus filtering out words that are syntactically similar but irrelevant.

Although we propose to use both of these methods in conjunction with each other to maximise the expansion of SenZi while maintaining high accuracy, we test each approach separately as well, using different embedding models and different numbers of nearest neighbours. We finally evaluate each resulting expansion of SenZi in Chapter 7 to draw conclusions.

We list the expansion approaches that we applied to SenZi below and follow by detailing each approach:

1. Word2Vec
2. FastText
3. CLS Matching
4. FastText + CLS Matching
5. FastText + CLS Matching Recursively
6. CLS Matching + FastText + CLS Matching

For each word embeddings model, we used the skip-gram approach since it works better with small data, rare words, and morphology (Bojanowski, et al., 2017).

Before we detail each approach, we note the following filtering steps we did during and after each expansion. First, for each SenZi words we only add a retrieved word to the vector of nearest neighbour if the word does not exist in that vector to avoid word duplicates within each vector. Second, after fetching all nearest neighbours, we sort all the retrieved words to facilitate the following steps:

1. Remove duplicated words, one or more, keeping one copy of each word.
2. Remove overlapping words, words that occur in positive and negative lists.
3. Remove non-alphabet words.
4. Remove words consisting of one letter.

We refer to these steps as *filtering* in describing each expansion below.

SenZi: 2K words (600 positive, 1.4K negative)

6.1.3.1 Word2Vec (SenZi W2V)

This model takes the position of words as parameters (Mikolov, et al., 2013). It does not consider the structure of words (no subword parameters). As mentioned earlier, unlike FastText, Word2Vec retrieves words that are related in meaning, words that co-occur frequently together. Irrelevant words (irrelevant in sentiment) such as neutral words co-occur naturally with sentiment words in text, as such, there is a high co-occurrence of neutral words or words with opposite sentiment using this model. For example, the mentioned word above *ndif - virtuous or clean* retrieved the following words.

ndif: ndif bidal fiyo ha2na fesid ntaha essmo ghayro terikho hayalla fechel ouat howee 3indo a3la tayaro cha2fit yente5ib tehdid byehterim

virtuous: virtuous, remains, within, our-right, corrupt, ended, his-name, other, his-history, whichever, loser, forces, him, with-him, higher, expelled, piece-of, he-votes, threat, he-respects

However, *wesikh - traitor, rude,* or *dirty* retrieved more relevant words, in meaning and sentiment.

wesikh: *wesikh elkaleb 7ayawen. wosikh ra2bit jabein khayin charafo kalebe kalb! monhat barbare ghachim nekir kheyen mret 7a2ir 5eyin 7a2oud 5erfen*

rude: rude, the-dog, animal, rude-form, responsibility, coward, traitor, his-honour, dog-feminine, dog, low-profile, barbaric, idiot, denier, traitor, useless, despicable, traitor, envious, insane.

We take this inconsistency into account and limit the nearest neighbours to 10, 20, and 50. We expand SenZi using each of these configurations and evaluate all three expansions in Chapter 7. We present these expansions in Table 6.2.

As can be seen, SenZi expanded from 2K words to 9.7K, 15.2K, and 25.3K using 10, 20, and 50 nearest word neighbours. We name this lexicon SenZi W2V.

	Expansion		Filtering		
Nearest Neighbours	Positive	Negative	Positive	Negative	Total
10	4.5K	8.6K	3.3K	6.4K	9.7K
20	8.5K	15.8K	5.3K	9.9K	15.2K
50	20.3K	37.5K	8.8K	16.5K	25.3K

Table 6.2: SenZi W2V, Word2Vec Expansions

6.1.3.2 FastText (SenZi FT)

FastText model (Bojanowski, et al., 2017) is an extention to Word2Vec (Mikolov, et al., 2013). In addition to the position of the words, FastText takes the word structure (subwords) as parameters. As mentioned earlier, the majority of the retrieved words are syntactically related. For example, the word *7abibi - darling* or *my-love*:

7abibi: 7abibo 7abibii 7abibe 7abibit 7abibak Hbibi 7abibeti 7abibiii 7abib 7abibeh 7abibet habibi 7abibt Habibi 7abiba 7abibty 7bibi 7abibna 7abi 7abibete

We increase the word neighbours to 50 and 100 to test if it still retrieves related words.

50 NN: *7abibo 7abibii 7abibe 7abibit 7abibak hbibi 7abibeti 7abibiii 7abib 7abibeh 7abibet habibi 7abibt habibi 7abiba 7abibty 7bibi 7abibna 7abi 7abibete 7abil khabibi s7abi 7abibto l7abib bibi habibi hbibi 7abibteh 7abibti 7bb 7abebet 7abebi habibi 7abait 7aboub 7abebe 7ammi 7abebti l7abi 7abb 7abayeb 5ayyi 7aby habibak 8ali 7abeeb 7obi habibii 7abebt*

100 NN: *7abibo 7abibii 7abibe 7abibit 7abibak hbibi 7abibeti 7abibiii 7abib 7abibeh 7abibet habibi 7abibt habibi 7abiba 7abibty 7bibi 7abibna 7abi 7abibete 7abil khabibi s7abi 7abibto l7abib bibi habibi hbibi 7abibteh 7abibti 7bb 7abebet 7abebi habibi 7abait 7aboub 7abebe 7ammi 7abebti l7abi 7abb 7abayeb 5ayyi 7aby habibak 8ali 7abeeb 7obi habibii 7abebt 5ayi 7abiss 7bbi habibii 7abak 7abboub 7anouni 7ami habibb hbb trekni jibi habibiii as7abi 7abit 5edni 7abeeet 7abeb 7amzi 7bib 5ayef mishta2lak 7amalak 7abel ma7rou2 habibiii habibit habibiiii sa7bi habibtak ma7ram 7abt habibe habibeh 3ayni 3eyni 7abei 7amada 7rub habib 7abten 7abibte 7obbi meshta2lak 7abasou habibik 7aiet ma7ru2 a5i romyi*

As can be seen the subword parameters by FastText retrieved more relevant words than the Word2Vec model. We observed this for several positive and negative words, thus based on this observation we retrieved 100 nearest neighbours for the fastText model. We present the 100NN fastText expansion of SenZi in Table 6.3:

Expansion		Filtering		
Positive	Negative	Positive	Negative	Total
58.3K	130.4K	10.3K	25.5K	35.8K

Table 6.3: SenZi FT 100NN Expansion

As can be seen, although 100 neighbours expanded SenZi from 2K to around 190K words, there was a large reduction during the filtering phase. The number of duplicated and overlapping words increase relatively with the size of SenZi. Nevertheless, this approach expanded SenZi to 35.8K words. We name this lexicon SenZi FT.

6.1.3.3 CLS Matching (SenZi Large)

As explained earlier, during our exploration of word embeddings expansion we heuristically found that if the nearest neighbours of a SenZi word contain the same sequence of consonant letters, then they are most likely to be forms of that word. However, since Arabic is a morphologically-rich language, where a triliteral stem could derive into many lemmas and inflections, this approach might match a high number of irrelevant words if ran against the entire vocabulary of the corpus. For that we favoured using it after retrieving a list of relevant words from an embedding model to limit the matching to the relevant words only. Nevertheless, we test this approach across the entire vocabulary of the corpus.

We list the vocabulary of the 1M Facebook corpus and remove all words written in non-Latin script such as Cyrillic, non-alphabet words, and one-letter words. This resulted in 892,169 unique words.

To decrease the error (erroneous matches), we set this condition: expand a SenZi word only if it contains a sequence of three consonant letters or more.

We iterate each SenZi word across all the vocabulary, normalising and matching in a hidden layer (detailed previously in Figure 6.5), to retrieve the matched words. However, to satisfy the three consonant letter condition we count the number of consonant letters post the normalisation to represent the number correctly. For example:

wesi(kh) has three consonant letters و س خ – *w s kh* with one represented as a compound letter *kh*.

wesikh ® *wskh* ® ws5: CLS 3
sharsha7a شرشحة - *very messy* ® shrsh7 ® r7: CLS 4

The word خير *kheir – good* for example has a CLS of size two *kh.r*. The sparsity of words matching this CLS is very high, such as:

kheir: khyar, kharma, kharouf, mkharaf, kharfen, khartesh, khartoushe, kharet, kharaz, kharze, kharab, mkharbat, khras, shakhir, sakhr, khardal, khere3, makhraj, khare2, kharej, kharjiye, etc..

kheir: cucumber, persimmon fruit, sheep, insane, insane, reload (bullet), bullet, chop or cheat, beads, bead, mess, confused, shut-up, snoring, rocks, mustard, weak, exit, infiltration, befitting, allowance, etc..

All of the mentioned words contain the same CLS as *kheir* but they are not semantically related. We can see the number of negative words matching with *kheir – good*. For that we keep short words expansion to the word embeddings. Below is the same word expanded in FastText.

kheir: kheir kheir bkheir lkheir elkheir kher kheirr kheir 5eir bikheir kheyr ekheir kheirrr khair 5er kher kher kheir l5eir khayr bkher b5eir 5eiir khayrat khere lkheyr gheir khetwe keir 5eyr khayra b5eyr lkher bkhayr khayran kherbi kherr lgheir

kheir: kheir-form, kheir-form, kheir-form, …, different, step

All of the nearest neighbours are very relevant, most of which are forms of *kheir* within the first 20 nearest neighbours at least. We present this expansion in Table 6.4.

SenZi, 2K words, expanded around 150 times in size using this approach. We name this lexicon SenZi Large.

Expansion		Filtering		
Positive	Negative	Positive	Negative	Total
226K	337.2K	125.1K	167.6K	292.7K

Table 6.4: SenZi Large, CLS Matching Expansion

6.1.3.4 FastText + CLS Matching (SenZi FT-CLS)

This is the approach described earlier in Figure 6.6. We used the FastText word embeddings model to retrieve related words then we applied CLS matching (Figure 6.5) to automatically select the most relevant words of SenZi, the ones that are potentially orthographic and morphological forms. Thus, limiting the CLS matching to the retrieved set of nearest neighbours. We present this expansion in Table 6.5.

Expansion		Filtering		
Positive	Negative	Positive	Negative	Total
7K	9.8K	6.4K	8.5K	14.9K

Table 6.5: SenZi FT-CLS, FastText Expansion with CLS Matching

Using the CLS matching with FastText model expanded SenZi from 2K to around 15K words, that is less than half the words without using the CLS matching (35K). We name this lexicon SenZi-FT-CLS.

6.1.3.5 FastText + CLS Matching Recursively (SenZi FT-CLSR)

This approach extends the previous FastText + CLS Matching with another round of expansion for each new relevant word. After the first expansion and CLS matching we take each new word, retrieve its nearest neighbours, and CLS match them with the original SenZi word, the parent word, for further expansion, visualised in Figure 6.7. We alse use the word RBZ to denote the CLS of ARABIZI, a SenZi word. We use $NN_{(NN)}$ to denote a new nearest neighbour of the first nearest neighbour.

We show the benefit of this approach through the following example.

tayab طياب means *cuteness or prettiness* in Lebanese Arabic. Retrieving 50 nearest neighbours using FastText + CLS expands this word to the following variants:

tayab: tayabb, atayab, ltayab, atyabek, tayob, atyabbb, atyabb, atyabooo, atyabu, atyaba, atyaboooo, 2tyab, atyaboo, atyabaa, atyabo, tayoub, atyab, atyabooooo, taybeee, atyabaaa, taybee, tayoubi, atybo, taybeeee, atyabaaaa, tayba, atybooo, atyba, tayben, taybii

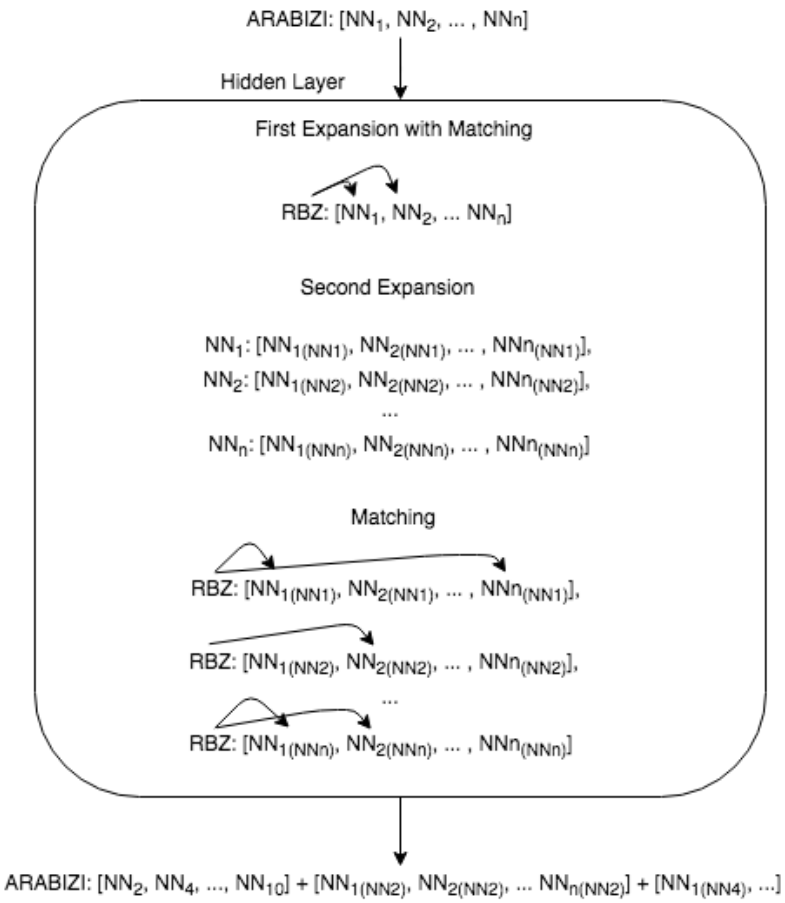


Figure 6.7: CLS Matching Recursively (Hidden Layer)

All of the retrieved words are forms of *tayab*. In the second round of expansion, we retrieve the nearest neighbours to each of these forms. For example, the 50 nearest neighbours expansion of the new word *atyab - how-cute* retrieves the following:

atyab: atyab, 2tyab, atyabu, 2atyab, atyabb, watyab, w2atyab, atyabo, atyaba, atyabek, atyabbb, tyab, atyabaa, atyabon, atyabou, atyb, atyaboo, atyabik, atyabaaa, atyaboun, atyabooo, 2atyaba, atyeb, atyabaaaa, atyabak, atyaboooo, atyba, atayab, atiyab, atyabooooo, atybo, atyabkoun, tayab, atyabkon, tyabo

Which updates the first retrieved list of words by seventeen new words.

2atyab, watyab, w2atyab, tyab, atyabon, atyabou, atyb, atyabik, atyaboun, 2atyaba, atyeb, atyabak, atiyab, atyabkoun, tayab, atyabkon, tyabo

Which includes inflectional forms of the newly expanded *atyab - how-cute*.

atyabak - how-cute-you-are (masculine)
atyabik - how-cute-you-are (feminine)
atyabon - how-cute-they-are
atyabkon and atyabkoun - how-cute-you-are (plural)

Although all new nearest neighbours (nearest neighbours of the first nearest neighbours) would be retrieved if we increase the number of nearest neighbours of the SenZi word in the first place without recursion, this approach focuses on the cluster of each word without the risk of retrieving as many irrelevant words. We visualise this in Figures 6.8 and 6.9. We add the recursion to the SenZi Expansion with CLS Matching diagram in Figure 6.10. We present this expansion in Table 6.6.

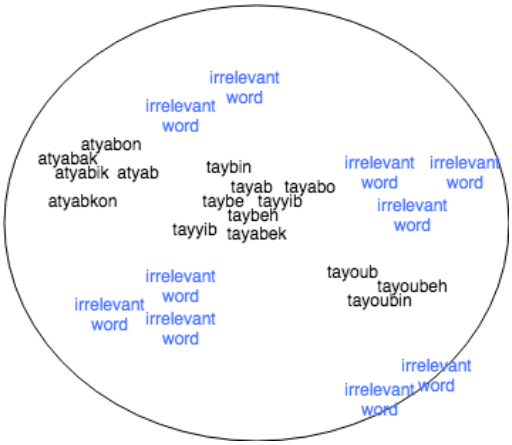


Figure 6.8: SenZi Expansion – Increasing Number of Nearest Neighbours

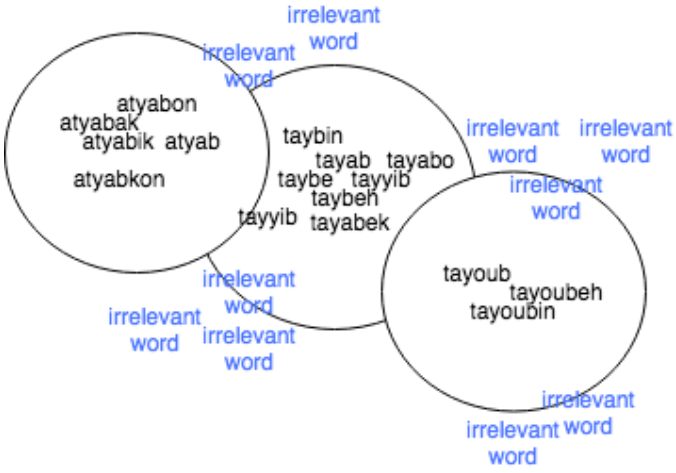


Figure 6.9: SenZi Expansion – Retrieving Nearest Neighbours Recursively

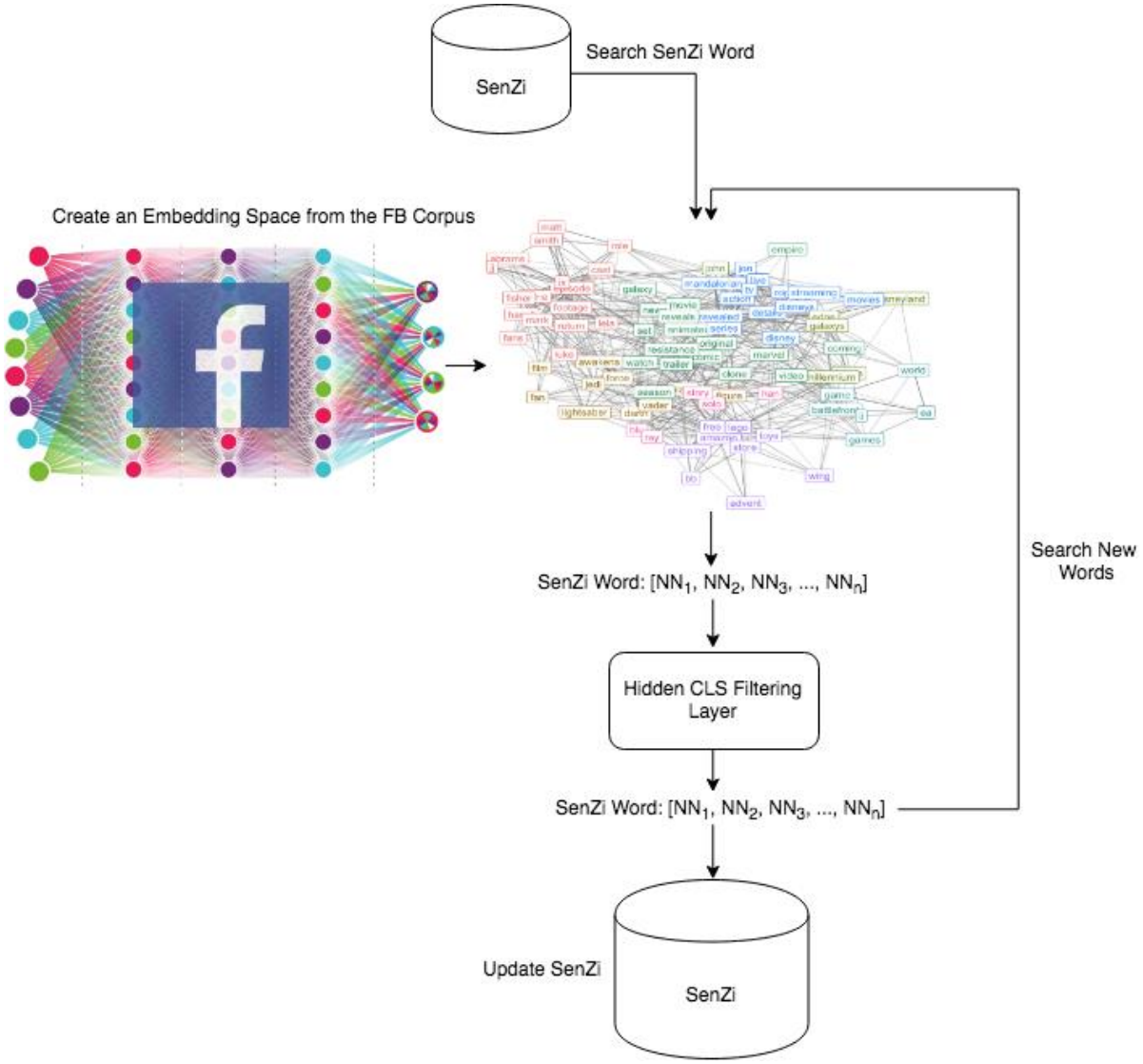


Figure 6.10: SenZi Expansion with CLS Matching Recursively

Expansion		Filtering		
Positive	Negative	Positive	Negative	Total
19.1K	24.6K	13.1K	14.8K	27.9K

Table 6.6: SenZi-FT-CLSR, FastText Expansion with CLS Matching Recursively

Adding the recursion expanded SenZi from 2K to 27.9K words, almost double the words over the previous expansion. We name this lexicon SenZi FT-CLSR.

6.1.3.6 CLS Matching + FastText + CLS Matching (SenZi Large-FT-CLSR)

In this approach we combine the third expansion that uses the CLS matching only with the fifth expansion that combines fastText with CLS matching recursively.

As mentioned in the third expansion, SenZi-Large, we limited the CLS matching expansion to SenZi words that consist of at least three consonant letters to reduce the errors. After observing the resulting lexicon SenZi-Large we noticed that majority of SenZi words matched with a high number of the word's forms, however each case is different. Words that consist of a unique sequence of consonant letters matched with a high number of its forms with minimum error and words that consist of a sequence of consonant letters that is common to other words matched with a high number of irrelevant words.

The word *wafa2* وفق *or* وافق for example, means *to wish good luck* or *agree on*, is the stem to many lemmas and inflections of the expression *Allah ywaf2 - May God brings good luck.* The CLS *wf2* is unique that it matched 913 related forms.

wafa2: 2eywaf2on, 3amtwafa2o, Beltewfi2, Beltiwfi2, Betawfi2, Brttawfii2, Bwef2aa, Bwef2ak, Bwefe2on, Lmouwefa2a, Lwefe2, Mouwefa2t, Mwaf2, Mwaf2a, Mwaf2in, Mwafa2, Mwafa2a, Mwafa2aaa, Mwafa2en, Mwafa2in, Mwafa2ine, Mwafaa2, Mwafeee2, Mwaffa2, Mwaffa2a, Mwaffa2iiiin, Mwaffa2in, Mwfa2, Tawafi2, Taweefou2, Tawefou2e, Tawf2na, Tawfi2, Twefa2na, Twf2na, Ywaf2o, Ywf2kooooooooooooon, Ywf2on, Ywffa2ak, ...

But in other cases, such as *jaben* جبان - *coward* contains a CLS of *jbn* which is common to other words. It matched 850 words with a high number of irrelevant words. We select some of the irrelevant words below:

jaben: 3ajaban, 3ajabna, 3ajbene 3ajbenik, 3ajbenk, 3ajbeno, 3ajbinak, Bey3jboune Estajbne, Istsjibna, Ljaben, Ljban, Mhajbin, Wajebna, Wm7ajbin, a3ajibon, eljaban, eljaben, eljbne, eljebne, eljiben, mit3ajbin, mnjebon, mnjibn, mnjibon, mo3ajbin, wejbin, wejbn, wejbna, wejbon, etc..

these words include forms of اعجاب عجب جبنة واجب جبين استجابة محجبة جيب - *to like something, having a crush over someone, cheese, obligation, weird, unusual, bring something, scarfed, forehead.*

We were not able to find a pattern to determine which CLS results in irrelevant words automatically without manual selection. This problem reflects the advantage of limiting the vocab to the neighbours of every SenZi word before the CLS matching. However, we noticed that words that consist of CLS of length four or more very rarely match irrelevant words. For example, *27tiram* احترام – *respect*, CLS: *7trm*, matched 1,871 forms that are relevant by apparent observation.

i7tiram: 27tiram, 27tarame, 27tarmi, 27teram, 27terame, 27terami, 27teramon, 27termo, 27tiram, 27tirame, 27tirameh, 27tirami, 27tiramon, 27tram, 27trame, 27trami, 27tramik, 27tramk, 27tramy, 27trme, 27trmek, 27trmi, 2al2e7tiram, 2e7tarami, 2e7teeram, 2e7teram, 2e7terama, 2e7teraman, 2e7terame, 2e7terami, 2e7teramm, 2e7teramna, 2e7teramo, 2e7terem, 2e7terim, 2e7terma, 2e7termak, 2e7tiram, 2e7tiramak, etc…

As such, in this expansion we take advantage of this approach to update the SenZi-FT-CLSR lexicon. We expand SenZi words that contain a CLS of at least four letters and merge this expansion with SenZi-FT-CLSR lexicon. We present this expansion in Table 6.7.

Words of CLS 4 or More		Expansion		Filtering		
Positive	Negative	Positive	Negative	Positive	Negative	Total
256	681	27.6K	41.1K	21.1K	35.1K	56.2K

Table 6.7: CLS Matching Expansion Limited to Words of CLS Length of 4 or More

This expanded SenZi from 2K to 56.2K words. We merge it with SenZi-FT-CLSR (27.9K words), keeping one copy of each word reaching a total of 80K words. We present this merge in Table 6.8. We name this lexicon SenZi Large-FT-CLSR.

SenZi FT-CLSR		SenZi Large-FT-CLSR		
Positive	Negative	Positive	Negative	Total
13.1K	14.8K	32.7K	47.2K	80K

Table 6.8: Merging SenZi Large (CLS 4 or more) with SenZi FT-CLSR

In the next chapter we evaluate and discuss each of these lexicons using lexicon-based approach against the annotated datasets prepared in Chapter 4.

6.2 Chapter Summary

In this chapter we addressed the high degree of lexical sparsity, a major challenge that would impede the coverage of sentiment words in the sentiment analysis of Arabizi.

We addressed this challenge by expanding the sentiment lexicon Senzi to include written forms of SenZi's words and their inflections. We used the large Facebook corpus created in Chapter 5 to retrieve the word forms and add them to SenZi. We utilised two approaches for this expansion:

1. Word Embeddings: Retrieves words that have semantic relationship with the input SenZi words.
2. CLS Matching: Matches all syntactically relevant words.

We used each of these approaches in different configurations, separately and together. We combined the two approaches to filter words retrieved by the word embedding models keeping the forms that are more likely to be relevant in orthography or morphology.

These approaches resulted in six new expanded versions of the original SenZi (2K words). We present a summary of these expansions in Table 6.9.

In the next chapter we fully address RQ3 to find whether word embeddings improve the sentiment analysis of Arabizi. We evaluate the original SenZi and each of its six expanded versions.

Expansion	Description	Size
SenZi W2V: 10NN	Word2Vec embeddings model . Retrieving 10, 20, and 50 NN. No filtering.	9.7K
SenZi W2V: 20NN		15.2K
SenZi W2V: 50NN		25.3K
SenZi FT	FastText embeddings model: 100 NN.	35.8K

	No filtering.	
SenZi Large	CLS matching with all words in corpus.	292.7K
SenZi FT-CLS	FastText embeddings model: 100 NN. Filter NN on CLS.	14.9K
SenZi FT-CLSR	FastText embeddings model: 100 NN. Filter NN on CLS. Repeat expansion and filtering for every new NN.	27.9K
SenZi Large FT-CLSR	CLS matching with all words in corpus for long SenZi words only. Merge with SenZi FT-CLSR.	80K

Table 6.9: Summary of SenZi Expansions

III. Sentiment Analysis

7 Evaluation

فما لي بعدٌ بعد بعدك بعدما تيقنت أنّ القرب والبعد واحد

So far we have created SenZi, a sentiment lexicon for Lebanese Arabizi, and expanded it using word embeddings, which resulted in six expanded versions of SenZi. This chapter answers RQ2 and RQ3 by evaluating SenZi and its expanded versions through sentiment analysis experiments.

RQ2: *How could an Arabizi sentiment lexicon be developed and used for sentiment analysis?*

RQ3: *Could word-embeddings enhance the performance of Arabizi sentiment analysis?*

Given the myriad number of challenges associated with Arabizi and the current scarcity of annotated data, we designate the lexicon-based approach as the evaluation method of SenZi sentiment classification presented in this chapter.

Knowing that Arabizi is also a low-resourced language, therefore building a sentiment lexicon and evaluating it using a lexicon-based approach is to our knowledge the first contribution to the sentiment resources of Arabizi, hence, a new baseline for researchers to build upon and benchmark future efforts in resourcing Arabizi.

In this Chapter we introduce the lexicon-based approach for sentiment analysis. We describe how we integrated SenZi in this approach to classify sentiment from the SA dataset created in Chapter 4 and detail the evaluation setups and experiment. We evaluated every sentiment lexicon we produced in Chapter 5 and 6.

After presenting the evaluations, we examine a portion of the classified twitter data to learn and present the factors that impact the lexicon-based sentiment analysis for Arabizi. We

finally discuss the major drawbacks of the approach and propose new research ideas to target.

7.1 The Lexicon-Based Approach

The lexicon-based approach is a relatively simple technique that scores tweets based on the occurrence of lexicon words in the text. It gives a score for every lexicon word found in the input text and aggregate these scores at the end to determine to which sentiment class the text belongs to, usually *positive, negative,* or *neutral.*

We show how the lexicon-based approach works and where it falls short in the following example, a comment taken from Facebook from a public page.

Figure 7.1: Facebook comment example

seriously he is shameless and "elo 3ein" / "dares to" (expression) talk we've become the most-rubbish country in the world and they are the ones that are not compliant.. "tfeh" / "shame" (disgusted expression) on such a government,,, we pray for the sleeping nation to wake up..

The approach reads the text, word by word, checking each word if it exists in the positive or negative list in the sentiment lexicon and scores each word according to the scores found in the lexicon. In our case it counts the positive and negative lexical words, and classifies the text positive if the positive words are greater than the negative words, negative if the negative words are greater, and no sentiment otherwise. This is equivalent to scoring positive words +1, negative words -1, and aggregating the scores at the end to classify the text.

Now let's assume that SenZi contained these negative words, *weki7 - shamless*, *azbal - most-rubbish*, and the disgusting expression *tfeh*. The approach would then classify this comment as negative.

Figure 7.2: Facebook Comment Example Classification

seriously he is shameless and "elo 3ein" / "dares to" (expression) talk we've become the most-rubbish country in the world and they are the ones that are not compliant.. "tfeh" / "shame" (disgusted expression) on such a government,,, we pray for the sleeping nation to wake up..

However, there are other sentiment features in the text that are difficult to classify.

1. *elo 3en – dares to: A* common negative expression lacking sentiment words.

2. *we pray for the sleeping nation to wake up:* This expresses negative sentiment towards the people of country without using negative words as well.

Hence such sarcastic texts bypass the lexicon-based approach.

Ideally, the desired outcome is not only detecting the *sleeping nation* but all sentiment features in the text:

seriously he is shameless and "elo 3ein" / "dares to" (expression) talk we've become the most-rubbish country in the world and they are the ones that are not compliant.. "tfeh" / "shame" (disgusted expression) on such a government,,, we pray for the sleeping nation to wake up..

Nevertheless, In the following subsections we detail the data preparation, lexicon based evaluation setup, present the results, and analyse the errors.

7.1.1 Data Preparation

We use the SA dataset created in Chapter 5 for the sentiment classification experiments. We now recap the creation of this dataset. We created this dataset from two connected annotation tasks: We asked three students to annotate 30K tweets. They checked whether each tweet is Arabizi, and for each tweet they identified as Arabizi, they were asked to annotate the tweet with a sentiment label: positive, negative, or neutral.

We took the tweets that at least two students agreed to be Arabizi and extracted the ones that at least two students agreed on their sentiment. This resulted in 2.9K Tweets: 801 positive, 881 negative, and 1.2K neutral. We perfectly balanced the data with 800 positive and 800 negative.

Prior to the annotation, we filtered out non-alphanumeral characters, urls, hashtags, and mentions to keep the tweets that are composed of words. We then deleted tweets that lack an alphabet and duplicated tweets. We did not preprocess the tweets any further to keep them meaningful for the students to read and annotate. However, after obtaining the annotation we have some room for light preprocessing prior to the sentiment classification experiments.

Some researches on sentiment analysis proposed heavy preprocessing of the input text before running the sentiment classification to reduce the degree of sparsity in a language such as lemmatization (Chapter 3).

Lemmatization is the reduction of words to their lemmas. In English for example, blindly trimming some suffixes from words in the input text simplifies the development of the lexicon with low risk of harming the data in this case such as:

enjoying ® enjoy
enjoyed ® enjoy
enjoyful ® enjoy
enjoyment ® enjoy

Hence, keeping one form of the word *enjoy* in the lexicon. This approach caters the data to match the resource, our approach is the other way around, we catered the resource to match

the data. We take this approach of handcrafting a sentiment lexicon and expanding it to cope with the aforementioned challenges of Arabizi, richness in morphology and inconsistency of orthography, since developing a lemmatizer or stemmer for any variant of Arabic is not as a straightforward trimming process as it is for English. The challenges are explained in Chapter 2. The value of our approach is creating a rich sentiment resource for Arabizi and keeping the preprocessing of the data to the minimum.

We apply a lighter form of preprocessing:

1. Simplify exaggerated words.
2. Remove stop words.

We simplified exaggerated words, words with repeated letters e.g. *love youuuu,* to reduce the sparsity even further. If a letter is repeated more than two times, we remove this repetition keeping one letter. For example *habibi – my-love*:

habbibiiiiii ® habbibi

The double *b* remains intact *ha<u>bb</u>ibi.* We keep double letters as this is common in Arabizi to express a shaddah phoneme, gemination (Chapter 2).

Although the lexicon contains exaggerated words after the expansion, there is still an endless space for exaggerating the text on social media.

Stop words are generally the most common words in a language. The idea of filtering stop words from texts is to keep the text to the words that matter to the classification task. In sentiment analysis, sentiment words, phrases, and expressions are the key features for the classification. Therefore, removing non-sentiment words that are common to a language such as linking verbs and prepositions automatically might facilitate the classification task (Saif, et al., 2014). For example:

I am a fighter for freedom, justice and for life
fighter freedom justice life

As can be seen the value of removing stop words depends on the task at hand. The remaining features in the filtered tweet suffice for plain sentiment classification, however, the value of the tweet degrades for entity and relation extraction. Who is *the fighter for freedom* is no longer detectable.

We create a list of stop words found in the Facebook corpus by getting the TF-IDF[36] scores for the words and selecting the words with the lowest score. TF-IDF stands for term frequency-inverse document frequency, it weighs the importance of each word in a document to the document, corpus in our case. The weight given to a word is proportional to the number of occurrence of the word multiplied by the inverse document frequency. This multiplication reduces the weight of the highly frequent words in a document such as the stop words. We selected 248 words from the lowest scored words. The list of stop words we obtained contains some negation words. We did not plan to filter the text from negation words as they play an important role in sentiment analysis. A negation could invert or diminish the sentiment of a sentiment word. As such we manually excluded negations from the list, resulting in 237 stop words presented in Table 7.1.

We created a filtered copy of the SA datasets, keeping both datasets, to test the sentiment analysis approach against both.

In the next section we describe how we detect negation followed by the evaluation setup.

7.1.2 Feature Extraction

As explained earlier, a lexicon-based approach classifies texts based on the occurrence of positive and negative words within the text. Before we run the classification we extracted one valuable feature from the text, negation, and integrated it in the approach. Other features include exaggeration, intensifiers, and emojis.

Intensifiers: *shu helwe - how beautiful*

[36] https://en.wikipedia.org/wiki/Tf-idf

wala ahdam - couldn't get any funnier

ktir fakhour - I am so proud

Emojis:

Exaggeration: wooooow 8eneye romanceyee - wooooow romanticc song

w	*ya*	*l*	*el*	*la*	*ana*	*bi*	*hal*	*fi*	*3a*
bas	*b*	*eh*	*al*	*men*	*3am*	*min*	*hek*	*bel*	*hayda*
3al	*shi*	*mn*	*li*	*kel*	*bl*	*chou*	*shu*	*akid*	*chi*
lal	*bs*	*eno*	*3ala*	*aw*	*eza*	*ken*	*ma3*	*be*	*law*
iza	*enta*	*3m*	*bass*	*sho*	*3an*	*wel*	*sar*	*hay*	*aya*
ta	*ra7*	*kil*	*ente*	*ba3ed*	*ha*	*heik*	*ba2a*	*ne7na*	*le*
leh	*lesh*	*hala2*	*abo*	*fe*	*3l*	*wa*	*enno*	*wl*	*a*
chu	*3*	*ento*	*no*	*fa*	*i*	*she*	*ya3ne*	*lek*	*ba3d*
ykoun	*hl*	*nehna*	*so*	*abel*	*ehh*	*il*	*hon*	*yali*	*yalle*
yale	*we*	*fiya*	*kam*	*7a*	*ela*	*the*	*fik*	*hol*	*ah*
h	*gher*	*mtl*	*plz*	*lk*	*hiye*	*hyda*	*ino*	*hene*	*keno*
ad	*elak*	*hk*	*3n*	*kell*	*kelna*	*lah*	*7ata*	*elo*	*am*
in	*et*	*is*	*to*	*y*	*kan*	*knt*	*kenit*	*kenet*	*tab*
wma	*ma3na*	*3la*	*ele*	*kter*	*ktir*	*2al*	*hadan*	*enti*	*la2an*
bado	*baddo*	*bade*	*hla2*	*it*	*ka*	*m3*	*cho*	*huwe*	*n7na*
saro	*3lek*	*yeh*	*me*	*enu*	*add*	*eli*	*3le*	*for*	*kmn*
kamen	*kamena*	*hayk*	*heye*	*meno*	*ktr*	*y3ne*	*mena*	*ila*	*wo*
inte	*bde*	*lel*	*sir*	*on*	*fee*	*hik*	*an*	*btw*	*3ndo*
haik	*kela*	*elle*	*e*	*ur*	*hel*	*or*	*3nd*	*mr*	*yes*
sarit	*ydal*	*nhna*	*hole*	*yeha*	*la2*	*u*	*at*	*my*	*but*
her	*kl*	*bil*	*with*	*halla2*	*this*	*ma3e*	*kello*	*and*	*mnl*
ade	*menne*	*enty*	*just*	*kelon*	*ekher*	*will*	*it's*	*amtin*	*elik*
la7	*then*	*mnel*	*tho*	*of*	*shou*	*yeje*	*mins*	*inta*	*ane*
tene	*ella*	*2aw*	*abl*	*hye*	*ye*	*inti*			

Table 7.1: Lebanese Arabizi Stop Words

Such linguistic features could have an impact on the sentiment classification, an increase or decrease in the positivity or negativity, or even help classify the text. However, we found it difficult to identify intensifiers and deal with Emojis.

The dialectal intensifier words *shu* and *wala* mentioned in the examples *shu helwe – how beautiful* and *wala ahdam – couldn't get any funnier* are contextual words that do not intensify in different context rather has totally different meaning *shu – what* and *wala – to swear or the preposition or.* Also, the intensified word could be before or after the intensifier, the example *ktir fakhour – I am so proud* may be expressed as *fakhour jiddan* where the intensifier is

positioned after the word *proud*. Taking this into account requires scrutiny of the linguistics of intensification in this dialectal Arabic. As mentioned in Section 3.3.3 we avoided handcrafting lots of rules and exemptions by all means.

We did not consider emojis to be cues for classifying sentiment text on social media, because in many cases they are used in sarcasm, such as using a smiley to express a negative sentiment.

fetna bl 7eit

we're screwed (expression)

As for the exaggeration, we cleaned the text from exaggerated words as mentioned earlier, but the question is *whether increasing or decreasing the score of an exaggerated positive or negative sentiment word improves the overall sentiment classification of the data*? We tested it to learn that it does not impact the overall sentiment classification in our case.

As such we overlooked these features and focused on what we believe has significant value to the sentiment classification, negation. Negation handling is wide subject in NLP, there is special focus on how the negation is used in the language and whether it negates or just lessens the polarity of the word (Liu, 2015). For example, if we consider the following examples in English (we use the term negators to refer to negation words).

not good: *not* inverted the sentiment, this phrase means *bad*.

not bad: Means it is *fine*, but not perfect.

not too bad: Here the negation is offset by a word between the negator and the sentiment word.

Similar to Arabic, a negator does not always invert the sentiment of a word and some negation is offset by a word or more, however we do not delve into negation handling thoroughly, rather we design a simple negation technique. If a negator occurs just before a sentiment word, we classify the word in its opposite sentiment class. For example:

بلا أخلاق *bala akhla2 - lacking good-morals*: If the word akhla2 - good-morals is in the positive list in the lexicon then the occurrence of this phrase will be classified as negative because the word *good-morals* is directly preceded by the negator *bala - lacking*.

We manually identified 11 negators from the list of 248 stop words:

bala, ma, manak ,mafi, mafik, mesh, mafesh, ma3ash, maba2, mar7, mal7

We expanded this list using the same expansion technique of SenZi, similar to the 5th expansion: SenZi-FT-CLSR. Recursive nearest neighbours retrieval with CLS matching using the same embedding space of the Facebook corpus. We obtained 167 negators presented in Table 7.2.

bala	*bla*	*m3ach*	*m3ash*	*m3ch*	*m3sh*	*m7a*
ma	*ma3ach*	*ma3ash*	*ma3ch*	*ma3sh*	*ma7*	*ma7a*
mab2	*mab2a*	*maba2*	*maba2a*	*mafch*	*mafe*	*mafech*
mafes	*mafesh*	*mafi*	*mafich*	*mafichi*	*mafie*	*mafih*
mafiha	*mafii*	*mafiii*	*mafiiii*	*mafik*	*mafiki*	*mafina*
mafine	*mafini*	*mafio*	*mafion*	*mafish*	*mafishi*	*mafiya*
mafiye	*mafiyi*	*mafiyo*	*mafiyon*	*mafsh*	*mafy*	*mal7*
mala7	*malah*	*mana*	*manak*	*mane*	*manha*	*mani*
manik	*manin*	*mank*	*manken*	*mankn*	*mankon*	*mankun*
manna	*mannak*	*manne*	*manni*	*mannik*	*manno*	*mannon*
mannoun	*mannu*	*mano*	*manon*	*manoo*	*manou*	*manu*
manun	*mar7*	*mara7*	*marah*	*marh*	*mb2a*	*mba2*
mba2a	*mch*	*mchi*	*mchn*	*mech*	*menak*	*menik*
menk	*menkn*	*menkon*	*mennk*	*mennkon*	*menno*	*meno*
menon	*mesh*	*mfch*	*mfech*	*mfesh*	*mfi*	*mfich*
mfish	*mich*	*miche*	*minak*	*minenne*	*minik*	*mink*
minkon	*minnik*	*mino*	*minon*	*mish*	*misha*	*mishh*
ml7	*mla7*	*mn*	*mna*	*mnik*	*mnk*	*mnna*
mnnak	*mnnik*	*mnnk*	*mnno*	*mnnon*	*mnnu*	*mno*
mnon	*mnu*	*mnun*	*moch*	*mosh*	*mouch*	*mr7*
mra7	*msh*	*mush*	*wbala*	*wma*	*wmaba2a*	*wmafi*
wmana	*wmanak*	*wmanna*	*wmanno*	*wmano*	*wmanon*	*wmara7*
wmarah	*wmch*	*wmech*	*wmeche*	*wmeno*	*wmesh*	*wmich*
wmino	*wmish*	*wmn*	*wmno*	*wmsh*		

Table 7.2: Lebanese Arabizi Negators

However, we added one exception to the negation technique. The word ما *ma* is contextual, a negator in some cases but an intensifier in other cases. For example:

ما أجمل السما *ma ajmal el sama - how beautiful the sky*

The word *ajmal* is the comparative form of the word *jamil - beautiful.* We learned heuristically that if the word ما *ma* precedes a sentiment word in its comparative form, it intensifies the sentiment. Given that the comparative form of words begins with the glottal stop phoneme ء transcribed as 2 or *a* in Lebanese Arabizi, we handled this exception accordingly.

The lexicon based approach is now loaded with the list of negators and ready to be loaded with different sentiment lexicons for sentiment analysis against the SA dataset. In the next section we detail the evaluation setup.

7.1.3 Evaluation Setup

The evaluation in this context is a measurement of how well the designated approach performs in classifying tweets into sentiment classes. This measurement is a direct comparison of the classification results with the humans' classification of the data.

The sentiment analysis approach we deploy is a two-class classification, *positive and negative,* a common approach in sentiment analysis where the classification is evaluated on how well it classifies positive and negative sentences (Nakov, et al., 2016). For that regard, we balance the SA dataset according to this type of analysis, 800 positive and 800 negative tweets.

As mentioned earlier, the lexicon-based approach matches the words in the tweets with the lexicon to classify the tweet. However, in two cases the approach does not classify tweets:

1. If the approach did not match any word with the lexicon.
2. If the positive and negative words are equal in a tweet.

With the present possibility of not classifying tweets we run two evaluations that deal with unclassified tweets differently.

In the first evaluation, we follow the method of (Al-Twairesh, et al., 2016), since the dataset is balanced between positive and negative tweets, we classify unclassified tweets as positive or negative randomly. We present the results of the lexicon-based classification using SenZi and its expansion. We then follow with a confusion matrix to show how frequent the approach fail to classify the tweets.

In the second evaluation, we first report the percentage of the classified tweets, then we present the sentiment classification results over the classified tweets. We also follow by presenting the confusion matrix.

Finally, we present a manual error analysis over the classified and the unclassified tweets to pinpoint the cases that bypass the lexicon based approach.

7.2 Results

7.2.1 First Evaluation

We randomised a class to unclassified tweets, hence every run of the experiment might produce a minor difference in the results. As such, we present the result of the lexicon based approach using SenZi and its expansions for three runs and average these results. We conducted this experiment against the SA dataset and a filtered copy of it, filtered from stop words, presented in Table 7.3.

As can be seen from the results, filtering the text from stop words had a miniscule impact on the sentiment analysis results. However, we consider the slightly better results from the filtered text for the following analysis.

The lexicon-based approach using the original SenZi achieved a 0.63 recall, 0.58 precision, 0.60 F1-score, and 0.58 accuracy. We expected the low results resulting from the high degree of sparsity in Arabizi, however, this proves that sentiment analysis on Arabizi text could be achieved without transliteration.

Each expansion of SenZi achieved a better F1-score than the original SenZi with SenZi FT-CLSR ranking the highest with a 0.76 recall, 0.66 precision, 0.71 F1-Score, and 0.69 accuracy pushing the results of SenZi original by a clear 13% in recall, 8% in precision, 11% in F1-score, and 11% in accuracy.

SA Dataset: 1.6K human annotated Arabizi tweets (800 positive and 800 negative)

	Unfiltered Dataset				Filtered Dataset			
	R	P	F	A	R	P	F	A
SenZi Original	0.54	0.59	0.57	0.58	0.69	0.57	0.62	0.58
	0.61	0.58	0.59	0.58	0.61	0.58	0.60	0.59
	0.59	0.58	0.59	0.58	0.60	0.58	0.59	0.58
Average	**0.58**	**0.58**	**0.58**	**0.58**	**0.63**	**0.58**	**0.60**	**0.58**
SenZi Word2Vec 10 NN	0.65	0.61	0.63	0.62	0.62	0.62	0.62	0.62
	0.60	0.62	0.61	0.62	0.60	0.62	0.61	0.62
	0.64	0.61	0.62	0.62	0.64	0.61	0.63	0.62
Average	**0.63**	**0.61**	**0.62**	**0.62**	**0.62**	**0.62**	**0.62**	**0.62**
SenZi Word2Vec 20 NN	0.67	0.61	0.64	0.62	0.57	0.63	0.60	0.62
	0.62	0.63	0.62	0.63	0.60	0.63	0.61	0.62
	0.68	0.61	0.65	0.63	0.66	0.61	0.63	0.62
Average	**0.65**	**0.62**	**0.64**	**0.63**	**0.61**	**0.62**	**0.61**	**0.62**
SenZi Word2Vec 50 NN	0.66	0.65	0.65	0.65	0.64	0.64	0.64	0.64
	0.63	0.66	0.65	0.66	0.63	0.65	0.64	0.64
	0.65	0.65	0.65	0.65	0.65	0.64	0.64	0.64
Average	**0.65**	**0.65**	**0.65**	**0.65**	**0.64**	**0.64**	**0.64**	**0.64**
SenZi FastText 100 NN	0.55	0.70	0.61	0.65	0.54	0.70	0.61	0.65
	0.57	0.68	0.62	0.65	0.63	0.66	0.65	0.65
	0.62	0.67	0.64	0.65	0.62	0.66	0.64	0.65
Average	**0.58**	**0.68**	**0.62**	**0.65**	**0.60**	**0.67**	**0.63**	**0.65**
SenZi Large	0.77	0.64	0.70	0.67	0.74	0.65	0.69	0.67
	0.69	0.66	0.68	0.67	0.67	0.66	0.67	0.67
	0.71	0.66	0.68	0.67	0.69	0.66	0.67	0.67
Average	**0.72**	**0.65**	**0.69**	**0.67**	**0.70**	**0.65**	**0.68**	**0.67**
SenZi FastText CLS 100 NN	0.69	0.68	0.68	0.68	0.71	0.66	0.68	0.67
	0.72	0.66	0.69	0.68	0.73	0.65	0.69	0.67
	0.72	0.66	0.69	0.67	0.74	0.65	0.69	0.67
Average	**0.71**	**0.67**	**0.69**	**0.68**	**0.73**	**0.65**	**0.69**	**0.67**
SenZi FastText CLSR 100 NN	0.74	0.67	0.70	0.68	0.79	0.65	0.71	0.68
	0.76	0.66	0.71	0.69	0.75	0.67	0.71	0.69
	0.76	0.66	0.71	0.69	0.73	0.67	0.70	0.69
Average	**0.75**	**0.66**	**0.71**	**0.69**	**0.76**	**0.66**	**0.71**	**0.69**
SenZi Large FastText CLSR 100 NN	0.76	0.61	0.68	0.64	0.76	0.62	0.69	0.65
	0.74	0.62	0.67	0.64	0.75	0.62	0.68	0.65
	0.72	0.62	0.67	0.64	0.69	0.64	0.66	0.64

Average	**0.74**	**0.62**	**0.67**	**0.64**	**0.73**	**0.63**	**0.68**	**0.65**

Table 7.3: 1st Evaluation on Unfiltered and Filtered SA datasets
Recall, **P**recision**, F**1-Score (Macro Averaging), **A**ccuracy

Although SenZi Large and Senzi Large-FT-CLSR also pushed the F1-scores of the original SenZi baseline by at least 8% they did not outperform SenZi FT-CLSR which indicates that expanding the words automatically to a large number of words introduces irrelevant words that harm the classification. As such, word embeddings using FastText model that takes the word structure as parameters and retrieve syntactically related words joined by the CLS matching of nearest neighbours recursively is the best expansion of SenZi using this evaluation approach.

In this experiment we randomised a class between positive and negative to the unclassified tweets. We now question what is the percentage of the unclassified tweets. We present the confusion matrices of the original SenZi and its best expansion SenZi FT-CLSR in Tables 7.4 and 7.5 to answer this question.

SenZi Original: 2K Words

Actual	Classified		Unclassified
	Positive	Negative	
Positive	20%	1%	79%
Negative	5%	20%	75%

Table 7.4: 1st Evaluation Confusion Matrix

SenZi FastText CLS Recursive

27.9K Words

Actual	Classified		Unclassified
	Positive	Negative	
Positive	56%	4%	40%
Negative	14%	39%	48%

Table 7.5: 1st Evaluation Confusion Matrix

The lexicon-based approach using SenZi original classified only 23% of the tweets. The classification increased to 56% using SenZi FT-CLSR. These results consolidate the argument that orthographic and inflectional forms play an important role in sentiment classification for Arabizi and potentially other morphologically rich languages that have inconsistent orthographies as well. We can also see that the error of classifying tweets in their opposite classes increased significantly after the expansion.

7.2.2 Second Evaluation

We now show how many tweets each lexicon classified out of the 1.6K filtered tweets dataset. Then we run the same lexicon-based approach over each set of classified tweets instead of assigning random classes to the unclassified tweets. We present the results in Table 7.6.

SA Dataset: 1.6K human annotated Arabizi tweets (800 positive and 800 negative)

	Unfiltered Data					Filtered Data				
Lexicon	Classified	R	P	F	A	Classified	R	P	F	A
SenZi Original	23%	0.95	0.81	0.88	0.87	23%	0.95	0.81	0.87	0.87
SenZi Word2Vec 10 NN	43%	0.80	0.77	0.78	0.78	43%	0.83	0.78	0.80	0.80
SenZi Word2Vec 20 NN	51%	0.78	0.73	0.75	0.75	49%	0.79	0.72	0.76	0.75
SenZi Word2Vec 50 NN	60%	0.76	0.74	0.75	0.75	57%	0.78	0.74	0.76	0.75
SenZi FastText 100 NN	69%	0.66	0.72	0.69	0.72	65%	0.69	0.76	0.72	0.74
SenZi Large	63%	0.84	0.73	0.78	0.77	63%	0.84	0.73	0.78	0.77
SenZi FastText CLS 100 NN	47%	0.95	0.83	0.88	0.87	47%	0.95	0.83	0.88	0.87
SenZi FastText CLSR 100 NN	56%	0.94	0.78	0.85	0.83	55%	0.93	0.80	0.86	0.84

SenZi Large FastText CLSR 100 NN	56%	0.91	0.69	0.79	0.75	54%	0.90	0.72	0.80	0.77

Table 7.6: 2nd Evaluation on Unfiltered and Filtered Dataset
Recall, **P**recision**, F**1-Score (Macro Averaging), **A**ccuracy

In this evaluation the results in the unfiltered text are slightly better. We therefore, consider the unfiltered text for the following analysis.

From these results we may draw several conclusions about the different methods of expanding SenZi. First of all, all expansions pushed the lexicon-based approach by at least 20% over the original lexicon.

Using the word structure as embedding parameters (FastText 100N) pushed the classification of tweets by 9% over not using the word structure as a parameter (Word2Vec 50NN), but did not improve the classification results, rather decreased by 10% in recall and 6% in F1-score. This implies that adding the subword information to the word position in the embeddings improves in retrieving syntactically related words, but worsens in semantically related words, different words but similar in meaning. Retrieving words that semantically related but not forms of SenZi words was not our goal in the expansion but proved to have good value in sentiment classification.

Similar to the first evaluation SenZi FT-CLSR achieved the highest results within 56% of the tweets. This lexicon was outperformed in classifying tweets by 13% difference from SenZi FT. However, SenZi FT-CLSR outperforms SenZi FT in classifying the tweets correctly by a significant difference of 28% in recall, 6% precision, 16% F1-score, and 11% accuracy.

SenZi FT-CLSR was also outperformed by SenZi Large in classifying tweets with 7% difference but similar to SenZi FT the classification results of SenZi FT-CLSR are higher by 10% in recall, 5% precision, 7% F1-score, and 6% accuracy. Merging SenZi Large with SenZi FT-CLSR had no improvement over SenZi FT-CLSR at all, not in classification nor in the classification results.

As such, the top expansions are SenZi FT, SenZi Large, and SenZi FT-CLSR. SenZi FT classified the highest number of tweets but achieved the lowest result among these three. SenZi FT CLSR on the contrary classified the lowest number of tweets achieving the highest results. SenZi Large achieved a more balanced number of classified tweets to classification accuracy over both SenZi FT and SenZi FT-CLSR.

SenZi Large expanded SenZi based on simple CLS matching; no word-embeddings were involved. This shows the power of this heuristic technique.

We list the confusion matrices for the mentioned top lexicons in Tables 7.7, 7.8, and 7.9 to examine how each lexicon performed in classification.

SenZi FastText: 35.8K words

Classified Tweets: 69%

Actual	Classified	
	Positive	Negative
Positive	32%	16%
Negative	12%	40%

Table 7.7: 2nd Evaluation Confusion Matrix

SenZi Large: 292K words

Classified Tweets: 63%

Actual	Classified	
	Positive	Negative
Positive	40%	8%
Negative	15%	36%

Table 7.8: 2nd Evaluation Confusion Matrix

SenZi FastText CLSR: 27.9K words

Classified Tweets: 56%

Actual	Classified	
	Positive	Negative

Positive	49%	3%
Negative	14%	34%

Table 7.9: 2nd Evaluation Confusion Matrix

This evaluation gave better understanding than the first evaluation where we randomised a sentiment class for the unclassified tweets. We now present an error analysis of the lexicon-based approach using the best performing lexicon SenZi FT-CLSR.

7.3 Error Analysis

Why is the lexicon-based approach failing to classify sentiment tweets?
Where does it go wrong in classifying positive or negative tweets?

In this section we aim to investigate the performance in the classified tweets and the reasons for not classifying the rest of the tweets. SenZi FT-CLSR classified 56% of the SA dataset which is 901 tweets leaving out 699 unclassified. We took a 10% random sample of the dataset to analyse the classification errors. We point out the cases that bypass the lexicon-based approach using SenZi FT-CLSR for the wrongly classified and the unclassified tweets.

Sample Data: 160 tweets.
Unclassified: 70 tweets.
Classified: 90 tweets (78 correct, 12 wrong).

We balance our analysis by analysing 12 tweets from each of these categories: Unclassified, classified correct, and classified wrong. Before we delve into the failures of SenZi, we analyse the successful classification to highlight the strengths of this approach and provide some insights. We present the tweets within each category, translate them, and highlight the words that were classified by the approach.

7.3.1 Correctly Classified Tweets

We observed that majority of the correctly classified tweets are simple unambiguous tweets that contained SenZi words. Some of which are too short that the sentiment word is obvious to determine the sentiment of the tweet as a whole such as:

1. *w ana b7ebbik*
   I-love-you too

2. *3ayb ya 3ame*
   shame man

3. *yii mabrouk*
   ohh congratulations

4. *habibi enti*
   you-are my-love

Other longer tweets also include one sentiment words that was a sufficient indicator to classify the tweet according to the class of that word:

5. *fi 3alam we27in la daraje*
   Some people are shameless to an extent (high extent)

6. *wlikkk nyiiiiiil alb kkl 7ada refa2ik bhl re7liii nyel albooooon*
   ohhh-youu "nyil alb" / "positive jealousy" (expression) *each one who companied you in this journeyyy "nyel albon"* (same expression different orthography)

7. *shouldve mentioned enak bayekh kamen*
   shouldve mentioned that you-are boring as-well

Some tweets consisted of several sentiment words, but matching one or two of them was enough to classify the tweets correctly:

8. *yaaaaaaay ana kteeer mabsouta w met7amsiiii wa akheran ra7 shufikkk bi wallah la nkayef* :-d
   *yaaaaaaay I am soo happy and excited**dd finally I will see-youuu surely we-will-enjoy* :-d

The words *happy* and *we-will-enjoy* matched with SenZi leading to a correct classification of the tweet, positive, although it missed *met7amsiiii - excited* in this exaggerated form.

SenZi also matched exaggerated sentiment words such as:

9. *yallllllla nwale3aaaaaaaaa*
   lets make-it-on-firreeee

Other tweets were classified based on irrelevant features, but they were classified correctly (lucky classification):

10. *hayda fans najwa karam bkaber l aleb*
    these fans of najwa karam "bkaber l aleb" / "makes one proud" (expression)

This tweet is labelled positive for the expression *bkaber l aleb - makes one proud*. The approach classified the tweet positive for matching the surname *karam - generousity* of the mentioned artist with SenZi. This also points out the limitation of SenZi in classifying common multi word expressions.

11. *ha tn2ote3 lkahrba*
    the electricity is about to cut off

This tweet was annotated as negative by the students because of the imminent event, the electricity is about to cut off. The approach classified it negative for wrongly classifying the word electricity negative. We traced the word electricity in SenZi to find that it came as nearest neighbour to a structurally similar word *kahrab - electrocuted.* A drawback of the automatic expansion of the sentiment words.

We finally learned that the lexicon-based approach theory, the number of positive and negative words present in text corresponds to the correct sentiment class, succeeds in classifying simple Arabizi texts such as the following example that includes a negation as well:

12. *hahahahaha man mech ma2boul ra7 ebke!!!! fakkaret zabbatouwa awwal chi 3emlouwa adrab! hahahahaha man it is not acceptable I am about to cry!!!! at first I thought they have fixed-it they've done it worse!*

    Positives: *fixed-it*
    Negatives: *cry*, *worse*, and a negated *acceptable*
    Sentiment Class: Negative

7.3.2 Wrongly Classified Tweets

We now dissect the 12 wrongly classified tweets in detail to present the limitations of the lexicon-based approach using SenZi. We point out to sentiment phrases and expressions, checked sentiment words that were unclassified or wrongly classified and traced, justified, and discussed each error.

1. *eh soukhafa la2an kelo te3weye mesh aktar lek sa2at l nizam wow so much freedom fi2 yeah they're silly because its all barking not more look the system fell wow so much freedom wake up*

    We start by looking into the unclassified negative words *soukhafa - they're-silly* and *te3weye - barking (degrading someone's words)*. We found that both of these words exist but in different forms in SenZi:

    soukhafa – they're silly is transcribed as *sokhafa* in SenZi, an orthographic difference. Although there are 96 forms in the cluster (nearest neighbours retrieved) of *sakhif - silly*.

    The closest form in structure to *te3weye* is *t3awe you* (masculine) *or she-barks*, an inflectional form of *te3weye - barking*.

    SenZi on the other hand classified the word *mesh - not* as positive. We found that it was retrieved as a nearest neighbour to the word *meshe – going well* for its structural similarity by FastText.

The classified word *nizam* has a contextual meaning, either *the system* in this case, or *good-order* for that it is in SenZi.

Finally, there are two issues in the last fragment:

wow so much freedom fi2
wow so much freedom wake-up

First it is codeswitched into English. Second, the sarcasm in *wow so much freedom* and *fi2 wake up*.

Two positive words are used in this negative expression, *wow* and *freedom,* and there is no sentiment in *wake-up* out of context. A good example that shows how the meanings of natural language are not bounded by a defined set of words. It also shows the edge of the lexicon based approach, even well designed lexicons are high likely to miss or confuse words taken out of their usual context for sarcasm.

2. *ntebhe w treke l telephone mn 2eedek. l 7ob ma byenfa3ek*
   be-aware and let the phone off your hand. the love won't benefit-you
   The word *ntebhe - be-aware* is found in SenZi for it is used in a common phrase *ntebhe 3a7alek* which expresses a positive wish *take care of yourself* but it means *be aware* in this context hence words in common sentimental expressions are not necessary positive or negative.

   The word *7ob - love* in its base form was not classified because it has been filtered from SenZi after removing the common words between the positive and negative lists. We checked that it was retrieved as a nearest neighbour to the negative word *a7be - whore* that has a similar structure.

   The word *byenfa3ek - benefit-you* (feminine) in the negated expression *won't benefit-you* is found in SenZi in different inflectional forms such as *byenfa3ak - benefit-you* (masculine). We checked such words where forms of the words exist in SenZi but not the words themselves if they exist in SenZi Large. We found that all three words, *byenfa3ek* from this example and *soukafa and te3weye* from the previous examples exist in SenZi Large.

However, they occurred with higher number of irrelevant words in their word clusters, for example *byenfa3ek* co-occurred with *tenfe3el* and *nenfe3el* forms of *provoke* for their CLS similarity with *byenfa3ek*. Such scenarios explain the better coverage of SenZi Large but lower accuracy over SenZi FastText CLSR in classifying sentiment tweets.

Let's assume that the sentiment words were correctly classified in this sentence:

l 7ob ma byenfa3ek
the love won't benefit-you

The approach would equate the number of positive and negative words thus fail to classify the tweet as negative. In this specific case not only *ma byenfa3ek - won't benefit-you* has a negative meaning but also it impacts the positive word *7ob - love* preceding it. This becomes similar in meaning to *love is useless* where the word *love* is no longer positive when compounded with such phrases. Hence, another example that shows the limitation of lexicon based approach in classifying natural language.

3. *allah la yreddik ya wayleeee*
   "allah la yreddik" / "negative wish" (expression) *"ya wayle"* (expression)

   The first expression is a negative wish that does not contain negative words. It is the combination of words that give a negative meaning. The impact of multi-word expressions (MWE) has on lexicon-based sentiment classification is apparent from this and the previous examples.

   The classified word in the expression *ya wayle* intensifies the negativity or positivity in a sentence or simply expresses a surprised or shocked feeling. However, it is most probably present in SenZi for its common use in expressing admiration as well such as *ya wayle ma ajmala - "ya wayle" how beautiful it is*. It is unclear at the moment whether the presence of contextual words in SenZi is an advantage or disadvantage. For example, SenZi is probably better with this word as positive if it is used way more in positive expressions and better without it otherwise. With the current scarcity of sufficient annotated corpora to determine the probabilities of contextual words as positive or negative it might be safer to remove such words from SenZi.

4. *hahah tla3 men rase*
   hahah get-out of my-head

   Similar to *mesh - not*, another stop word *men - of* or *from* was classified as positive. This word was retrieved as a neighbour to *imen – strong faith.*

   Similar to the previous issue with the expressions as well, the word *rase - my-head* is part of a very common Lebanese expression على راسي *3ala rase* which literally translates to *on-top-of my-head* meaning *you or your words are so valuable to me* or *you're welcome*, a way of showing respect.

5. *ya khayne wen yale bado yekhedne coffee date?*
   you cheater where is the one who wants to take-me for a coffee date?

   Similar to previous errors, the first is not classifying the word *khayne - cheater*, the second is classifying *yekhedne - take-me* falsely. *khayne* was filtered from the negative list because it overlaps with the same word in the positive list. It was retrieved as a nearest neighbour to *khaye - my-brother* which is used in positive contexts. Similarly, the word *yekhedne - take-me* was retrieved in the positive list near the word *hedne - truce*. As can be seen from this and previous examples, a weakness in SenZi results in a direct misclassification of tweets.

6. *lee 2atesh rassak b ur avi on snap ahla l soura*
   why did you chop your-head in ur avi on snap the (original) picture is nicer

   In this tweet the form of *ras - head* is also wrongly classified as it was retrieved from the word *my-head* which is part of a positive expression *3ala rase* explained earlier, however this did not impact the tweet as a whole to be classified as positive due to the occurrence of another positive word nicer. This tweet is wrongly classified because it did not match the annotation of the students (negative), although in our opinion the author of the tweet is describing the picture as nicer. This shows that the evaluation data has room for imprecise annotation and difference in opinion.

7. *mshi in 5 mins w bala na2*
   move in 5 mins and don't nag

   Similar to *mesh, mshi - move,* also a nearest neighbour to *meshe - going-well. na2 - nag* is not in the lexicon in this form though *na2a2 - nagger* and other inflectional forms are in the lexicon. We notice here that the word preceding *na2* is *bala,* a negation that is used in the imperative case in this example. Though it is common to negate words such as *bala akhla2 - lacking morals,* negating the negative word *nag* inverts its polarity falsely. As such, even negation words are contextual.

8. *shu 3amlitla enti? 2ooli mesh tal3ini b swad l wej atla3 men taraf l ghaltan*
   what have you done to her? say don't "tal3ini b swad l wej" / "embarrass me" (expression) *I'll end up coming from the wronged side*

   This tweet begins with the question:

   shu 3amlitla enti?
   what have you (feminine) *done to her?*

   Which does not contain common expressions nor sentiment words yet it imposes negativity. This example raises another challenge, interrogative sentences (Liu, 2015). Interrogative sentences may contain sentiment words but not the sentiment, or lack sentiment words but impose a sentiment like the mentioned tweet. For example:

   heyda istez mni7?
   is he a good teacher?

   The positive word *mni7 - good* is not a confirmed attribute of the teacher.

   shu khassak?
   how are you involved?

   This question means *it is none of your business.*

Expressions pose the same challenge to sentiment analysis. The common expression *tal3ini b swad el wej* means *make me look bad in front of others* has only a contextual sentiment word *swad – blackened* used for negativity.

The negative word *ghaltan - wronged* was not classified because it was filtered from the negative list due to its appearance in the positive list. It was retrieved as a neighbour to the word *ghale - highly-valuable,* a word that is contextual in the first place, it could mean *expensive*.

The classification of the word *men - from* is explained in a previous example. As for the word *taraf - side,* it is classified as negative because it is retrieved as a neighbour to the word *araf - disgusting* that has a similar structure.

9. *elet badde rayyih 3youne 30 minutes ta oum mnashta rehet nemet se3a w nos*
   I thought I'll rest my-eyes 30 minutes to wake-up energetic I went to sleep for an hour and a half

   The same types of errors are re-occurring; both words *rayyih - rest* and *mnashta - energetic* (feminine) do exist in SenZi as they are but they do in different orthographic and inflectional forms. We note that both of these words *rayyih* and *mnashta* are used in more positive forms such as *mraya7* or *merte7a - chilling* or *relaxing* and *nashit - active* or *in good health*. The word *3youne - my-eyes* is a retrieved form of the word *3ayne* that has a literal meaning *my-eye* but used positively as *dear, darling,* or *my-love*.

   This tweet proves again that analysing sentiment in natural language could not be limited to a list of words. Now let's assume that the approach classified these words correctly, *rest* and *energetic,* it will falsely classify the tweet as positive. In this tweet the negative sentiment is a disappointment tied with the time one hour and a half. The positive sense *wake-up energetic* failed to happen. Capturing such meanings is beyond the lexicon-based approach (Liu, 2015).

10. *kif elik 3ein tou2afe barra natra wahad? wen karamtik???? khalli houwe yelha2ik*
    how "elik 3ein"/ "dare you" (expression) *stand outside waiting for a guy? where is your dignity??? Let him chase you*

The first expression *kif elik 3ein* meaning *how dare you* translates literally to *how do-you-have an-eye*, thus lacking sentiment words. The word كرامة *karama - dignity* is considered positive in Arabic for its *generosity* and *honour* meanings. However, it is used in the interrogative case in this example *where is your dignity?* in other words *do you lack dignity?* as such this introduces the possibility that interrogative sentences negate sentiment words as well.

11. *tkheyal... bro beseer wejje asfar w akhdar*
    imagine bro my face will turn yellow and green

    tkheyal - imagine was retrieved as a neighbour to *khaye - my-brother* which is used in positive contexts as explained in an earlier example. The neutral adjectives *yellow* and *green* were used in the phrase to express a negative feeling.

12. *i know bas b awal l game it was fine. ma fi spirit bil marra bl team hayda chi wadi7 ma bada tnen ye7ko fiha*
    i know but in the beginning of the game it was fine. no spirit at all within this team its obvious "ma bada tnen ye7ko fiha" (affirmation expression).

    This tweet shows that sentiment analysis is beyond a simple polarity classification of positive and negative words. The author of the tweet described a game that it was fine, they codeswitched to English to express this positive opinion. Then they expressed a negative opinion towards the team's performance mixing Arabizi and English as well *ma fi spirit - lacking spirit* and *bil marra* / "horrible" (expression). This tweet presents a transition in sentiment from positive to negative. A simple approach to capture this shift in sentiment is to detect features such as sentence connectors *but, this, then, however, although etc.*. (Liu, 2015) in Arabizi *bas and ba3den* however the change in sentiment in this tweet was in two separate sentences without a connector. This tweet also shows that Arabizi users not only codeswitch frequently with English but some codeswitch in expressing sentiment words.

    Arabizi falsely classified *wadi7* as positive which means *obvious* in this case but also *vivid, clear,* or *understandable* in other cases.

After dissecting and analysing the wrongly classified tweets and discussing the errors, we summarise the types of errors, presented in Table 7.10.

Error	Description
Unclassified-New	Does not exist in SenZi in any form (new word).
Unclassified-Different	Exists in SenZi but in different form (orthographic / inflectional).
Unclassified-Filtered	Filtered from SenZi (overlap between positive and negative lists).
Stop word	Classifying a stop word.
Wrong NN	Classifying a word that it is an irrelevant neighbour to a SenZi word.
Contextual word	A SenZi word that has several meanings.
Part of expression	A SenZi words that is part of a common sentiment expression.
Sarcasm	Sentiment words for sarcasm.
Expression	Common expressions that present sentiment without sentiment words.
No sentiment words	Sentences that present sentiment without sentiment words.
Interrogative	Sentiment words that lose their sentiment when used in question forms.
Codeswitching	Sentiment words in English.

Table 7.10: Table of Errors Found and their Description

In the next set of tweets, the unclassified tweets, we translate the tweets and write the unclassified and wrongly classified words as well, however for ease of analysis, we label each of these words with their corresponding label from Table 7.10 and raise new errors if they occur.

7.3.3 Unclassified Tweets

There are two types of tweets in this category: Tweets that the approach classified an equal number of positive and negative words and tweets that the approach did not classify any words.

1. *ya 7abibetna enti sourtik bi albna wayn ma tkouni*
   oh our-beloved your picture is in our-heart wherever you are (feminine)

7abibetna - our-beloved: Unclassified-Different.
bi albna - in our-heart: Expression.

2. *lahza hayda l cutie elik?*
   Is this cutie yours?
   cutie: Codeswitching.

3. *kteer betawattar bas koon aam edfaa lal jemaa online aashen bhess ha y2orto aaleye kel musreeyete.*
   I get so nervous when I am paying the tuition fees online cause I feel they will nick all my money

   betawattar - I-get-nervous: Unclassified-Different.
   y2orto - they-steal (vulgar): Unclassified-Different.

   Although Arabizi has inconsistent orthography to be judged, there is a typo in the word *betawattar - I-get-nervous* that changes the usual pronunciation of the word *betwattar*. We add Typos to the list of errors.

4. *lak shou hal sowarrrrre w shou hal lookssss hawdeeeeeeee ???????????????? amaaarrrr*
   What are these pics (exaggerated) *and what are these looks* (exaggerated)*??.. beautiful*

   shou hal looksss - what are these looks: Codeswitched and No Sentiment Words.
   amaarrr - the moon: Expression.

5. *3a2ases bi waselne 3al sheghel bas kabne 3al tari2 w alle ekhod service*
   He was supposed to drive me to work but he-threw-me on the road and told me to take a cab

   kabne - he-threw-me: Contextual (could mean *drop-me*).
   he-dropped-me on the road and told me to take a cab: No sentiment words

6. *wooooow 8eneye romanceyee wooooow ya najwaa ataltene bhl 5abreyee natren 3ala nar l 8eneyee*
   wow (exaggerated) *romantic* (exaggerated) *song wow* (exaggerated) *oh najwa (artist) you-killed-me* (expression) *with this news we-are-waiting the song on-fire* (expression)

woooow: Codeswithcing (used twice)

romanceyee - romantic: Unclassified-Different.

ataltene - you-killed-me: Unclassified-Different (negative in SenZi), Contextual, Expression.

3ala nar - with lots of excitement: Expression.

7. *ma fike ta3mle fina hek !!!*
   You can't do this to us !!!

   No Sentiment Words

8. *70 slides to go abel bukra w ma baaref aan chu byehko tbfh.*
   *70 slides to go before tomorrow and I don't know what they are about tbfh (to be f***ing honest)*

   I don't know what they are about: no sentiment words.

   tbfh: codeswithcing, abbreviated English expression.

9. *bheb wajjeh tahiyye lal dekene l btdall fetha lal se3a 10 bl day3a*
   I would-love to send my appreciation to the shop that remains open till 10 in our village

   tahiyye - appreciation: Unclassified-Different.

   day3a - village: Polysemic word *village / confused or lost* (feminine).

   This polysemy is a result of the inconsistent orthography. These are two distinct words in Arabic ضايعة and ضيعة one with long vowel and one with short vowel (originally a diacritic). Since there is no differentiation between long and short vowels in Arabizi transcription, both words are transcribed the same *day3a* mentioned in Chapter 2. We add Polysemy to the list of errors.

10. *bahhaaahhahhah bass sha3re mish 2asir. 7atta hay w mish zabta*
    bahahahahahahha but my hair is not short. Even this one is not appropriate (expression)

    mish 2asir - not short (negated negative): Wrong NN

11. *haha ze2 mratab allah ykassir 2ide kif ken elo aleb ?*

haha decent taste (typo: zw2) *hope he breaks his hands how did he have a heart* ?

mrattab - decent: Sarcasm
have a heart: Expression.

12. *x at 3:30 a.m: i miss you. me: yii bel sharaf eh kifon ahlak w hek?*
x at 3.30 am: i miss you. me: ohh honestly yeah hows your-parents and stuff?

miss you: codeswitching
bel sharaf - honestly: Wrong NN.
ahlak - your-parents: Wrong NN.
kifon ahlak w hek yeah - hows your-parents and stuff: No Sentiment Words (Sarcastic).

In this analysis most errors belong to the table of errors except for two new errors: Typos (Tweet 3) and Polysemy (Tweet 9). In the next section we present the distribution of error percentages.

7.3.4 Results

In Table 7.10 we summarised the types of errors that occurred in the wrongly classified tweets. We covered errors that relate to words and errors that relate to sentences. We now present the percentage of each error on the word level and sentence level separately.
In this analysis we give attention to the classification of words, not the final tweets. Since in most of the cases classifying words correctly results in correct tweet classification, analysing the pitfalls of the word classification to understand the error and propose solutions is a contribution to the Arabizi sentiment analysis as a whole.

For a total of 24 Tweets there was a vocabulary of 178 words excluding short words that consist of one or two characters only and non-alphanumeric words. Out of the 178 words there were 30 sentiment words, of which 6 were classified correctly and 24 unclassified. Apart from the 30 sentiment words there were 19 wrongly classified words. We present the percentage of each error within the unclassified sentiment words in Table 7.11 and the wrongly classified words in Table 7.12.

As for the sentence level analysis, out of 24 tweets there were 32 sentences of which 19 (60%) presented sentiment without sentiment words such as expressions.

Although the sample size is small, this scrutiny of errors helped us identify the major drawbacks of the lexicon-based approach using SenZi for Arabizi sentiment analysis. In the next section we discuss these drawbacks and propose some research directions to address them.

24 Unclassified Sentiment Words

Error	Percentage
Unclassified-Different	46%
Code-Switching	37%
Unclassified-Filtered	13%
Typo	4%

Table 7.11 Error Distribution in Unclassified Sentiment Words

19 Wrongly Classified Words

Error	Percentage
Wrong NN	42%
Part of Expression	15%
Stop Words	11%
Contextual Words	11%
Sarcasm	11%
Interrogative	5%
Polysemy	5%

Table 7.12 Error Distribution in Wrongly Classified Words

After evaluating the sentiment lexicons using the lexicon-based sentiment analysis approach and reporting the cases that bypassed the classification, we now extend the results and error analysis with a further investigation of the drawbacks and propose new ideas to address them.

7.4 Discussion

In this discussion we refer to the unclassified sentiment words in Table 7.11 and the wrongly classified words in Table 7.12.

First of all, among the unclassified words category, there was no Unclassified-New errors which are sentiment words new to SenZi, not found in any form. Rather the majority of the unclassified sentiment words are Unclassified-Different errors that are sentiment words either inflected or transcribed differently from the original words in SenZi. Although we have expanded SenZi by around 15 times in SenZi FT-CLSR with 29.7K words. This highlights the magnitude of the high degree sparsity problem in Arabizi.

The second most occurring error in this category is the Code-Switching, where sentiment words are expressed in English. Arabizi in nature contains codeswitching with other Latin script languages; apparently English is entrenched in the Lebanese Arabizi.

On the other hand, the Wrong-NN made up the majority of the wrongly classified words which are irrelevant words that were retrieved as nearest neighbours in the automatic expansion of the SenZi words. This aligned with our intuition, increasing the coverage introduces irrelevant words. However, we noticed that majority of the Wrong-NN errors are words that consist of two consonant letters. For example:

mesh (m.sh), rase (r.s), men (m.n), mshi (m.sh)

This goes back to the same observation in Chapter 6 that motivated us to create the last version of the expanded SenZi, SenZi Large FT-CLSR. We limited the CLS expansion of SenZi to the words that consist of four consonant letters or more after noticing that words of two consonant letters and some of three consonant letters retrieved many irrelevant words. We merged this expansion with SenZi FT-CLSR where all words were expanded regardless of their CLS size from the embedding space. We now learn that even the embedding expansion of FastText with CLSR of short words harmed SenZi with irrelevant words.

In short, the best scoring expansion of SenZi was SenZi FT-CLSR, an expansion from 2K words into 27.9K. This automatic expansion raised the classification of tweets from 23% to

56% shown (Table 7.6), a clear advantage of expanding SenZi automatically to address the challenge of high lexical sparsity in Arabizi at the expense of introducing irrelevant words that lead to wrong or misclassification of tweets. The following observations sum up the majority of the errors in the SenZi lexicon-based classification of Arabizi tweets.

1. Not every form of SenZi words has the same sentiment of that word.
2. There is a high frequency of codeswitching to English in Lebanese Arabizi that impacts the sentiment classification.
3. Sentiment is not necessarily expressed in sentiment words, rather many sentiment phrases lacked sentiment words.

We now present our suggestions to deal with the mentioned observations.

7.4.1 Irrelevant Nearest Neighbours

One way to address the first observation is to give each sentiment word a polarity score. Besides being positive or negative, each word can have a positivity or negativity score such as 0.56 or -0.76. This approach gives higher value to words that dominate the sentiment over words that slightly impact the sentiment of the text. As a result, an equal number of positive and negative words present in some text do not necessarily equate each other in sentiment value.

The requirement to achieve this scoring for Arabizi, taking the sparsity into account, would be a large annotated dataset. The probability of sentiment words occurring in positive or negative text dictates the polarity scores of the words. With the inconsistent orthography and rich morphology in Arabizi, large annotated datasets are required for calculating the probabilities of sentiment words. This procedure of annotating datasets is costly in terms of time and annotation, as can be seen in Chapter 4, out of 30K tweets there were 3.4K Arabizi of which 801 are positive, 881 negative, and 1.2K neutral. However, the outcome resources of this thesis might be utiltised to reduce the cost of a new Arabizi sentiment annotation to create datasets for scoring SenZi or other Arabizi sentiment lexicons.

The Arabizi identification identifies Arabizi text from other Latinscript languages and the lexicon-based approach using SenZi classifies 56% of sentiment tweets with a 0.85 F1-score at its best. Hence both of these resources can be utilised for reducing the cost of creating new datasets:

1. Using the Arabizi identifier to prepare Arabizi texts for sentiment annotation reducing the time to identify the Arabizi sentences among Latinscript texts.

2. Using the lexicon-based approach with SenZi to classify new data into sentiment classes to prepare it for annotation reducing the neutral text thus increases the number of positive and negative texts in the dataset.

Another way of addressing this error is to filter irrelevant word neighbours from each SenZi word. The most accurate way is to do it manually at a very high cost. This also limits SenZi from being upgraded easily. In the current setup of the lexicon, the manual work takes place in the generation phase of SenZi while the expansion is fully automated. In this way, SenZi is easily maintained and expanded.

We propose an approach to filter the irrelevant words automatically by finding a relation metric between the retrieved relevant words and the irrelevant words within a cluster of words.

A metric based on linguistic patterns is very challenging because the irrelevant words retrieved have a very similar word structure such as the ones mentioned in the examples: *araf, taraf / disgusting, side* and *khaye khayne / my-brother, cheater (feminine)*.

A semantic relation metric among the words seems to be the most plausible. As discovered in Chapter 6, the linguistic CLS matching of words retrieved many forms of SenZi words that were not retrieved as nearest neighbours in the embedding space. This opens a new research direction to study why the vectors of these forms did not cluster with vectors of the SenZi words in the embedding space. Conducting this research involves accessing the internal structure of the word embedding models and tuning its parameters.

7.4.2 Codeswitching

Arabizi users from Lebanon constantly codeswitch with English reflecting their bilingual speech in text. The level of codeswitching could differ in different regions and dialects as shown in the pilot study in Chapter 2. With 37% of the unclassified words written in English, sentiment analysis for Lebanese Arabizi is incomplete without accounting for sentiment words in English. This challenge could be addressed by integrating an English sentiment lexicon in SenZi. This approach requires a linguistic study on codeswitching in Lebanese Arabizi that covers:

1. How frequent is codeswitching in Arabizi.
2. In which contexts users codeswtich to English.
3. Which sentiment words are expressed in English.
4. How are the English sentiment words used in Arabizi (exaggeration, typos, etc..).

The datasets provided in this thesis could be used for such linguistic studies.

Integrating an English sentiment lexicon in the lexicon-based approach requires a slick insertion into SenZi to avoid overlapping words with Arabizi. For example:

English *admin (administrator)*
Arabizi *admin: I-get-addicted,* a dialectal inflection of the word addiction ادمان *edmen.*

This becomes harmful if a sentiment word in the English lexicon overlaps with an Arabizi word of the opposite sentiment class. For example:

English *chum*: An intimate friend or companion.
Arabizi *chum,* an orthographic form of *shoum, shum, choum, chum,* etc.., meaning *shame,* also a part of a common negative phrase يا عيب الشوم *ya 3ayb el shoum* derived dialectally from the word شؤم which means the *evil consequence* or *misfortune*. It is normal for the *sh* phonetic to be expressed as *ch* in Arabizi, possibly originating from Arabizi transcription that came from users whose French is their second language.

Another important aspect in integrating an English sentiment lexicon is the form of English used in Arabizi. If jargon is the English used in Arabizi such as the mentioned *tbfh (to be*

*f***ing honest)* then a formal English sentiment lexicon might not add value to SenZi for Lebanese Arabic sentiment analysis.

Finally, an analysis useful for handling codeswitching in Arabizi is to check which Arabizi words often co-occur with English words. Maybe there is a common pattern in codeswitching among Arabizi users coming from the same region. An interesting way of conducting such a study would be to create a codeswitched corpus and train it in word embeddings to observe which Arabizi words neighbour the English jargon sentiment words.

7.4.3 Lack of Sentiment Words

A lexicon-based approach is a word-based sentiment classification. As seen from the examples, positive and negative sentiment is not limited to individual words rather 60% of 32 sentences found in 24 tweets express sentiment without sentiment words. Such as *tla3 min rase - get out of my head* or *kif elik 3en – how dare you.*

Word-level lexicon based sentiment classification scratches the surface of sentiment analysis. The complexity of multiword expressions (MWE) multiplies with the inconsistent orthography present in Arabizi. We take the two words *elik 3en* from the expression *kif elik 3en – how dare you* as an example to demonstrate this complexity.

Let's assume that each of these words could be written in any of the below orthographies:

elik 3en

elik: elik, 2lk, 2lek, 2lik, ilik, ilek, ilk, elek, elk	8 Forms
3en: 3en, 3ein, ein, 3ain, 3een, 3eyn, 3ayn, 3yn	8 Forms

As such this expression can be transcribed in at least 8x8 different orthographies. The phrase in this example *dare you* (feminine) could be used in different inflections such as the masculine and plurals forms. We present the different inflections of the word *elik - your* below:

elik: you feminine

elak: you masculine
elkon: you plural
ela: her
elo: his
elon: their
elhon: their

If each of those inflections can be written in 8 different orthographic forms then the expression would have at least (7x8)x8, 448 forms. In Figure 7.3 we present a snapshot of a Facebook comment that has the mentioned expression *how dare you* in a different form posted in reply to a public news post about a parliament convention that was about to take place in Lebanon (2019).

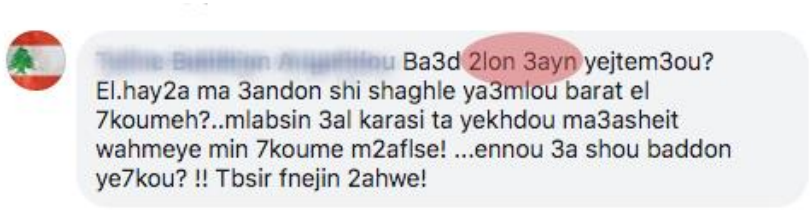


Figure 7.3: Facebook Comment Example

do they still dare to meet? It seems that they have nothing to do outside the government?..they are sticking on the chairs to receive imaginary incomes from a bankrupted government! … what is it they want to talk about? !! Coffee cup reading! (Fortune telling)

In this post the user wrote *ba3d 2lon 3ayn yejtem3ou? – do they still dare to meet?* using the plural form *their* of this expression in the certain orthographies *2lon 3ayn* mentioned in the list of forms above.

First, there are two types of sentences that express sentiment but lack sentiment words:

1. Common multi-word expressions.
2. Simple natural language that include hate or appreciation.

All five sentences in this comment express negative sentiment. They all lack sentiment words except the second one:

they are sticking on the chairs to receive imaginary incomes from a bankrupted government.

The expression *2lon 3ayn – still dare* in the first sentence is a common expression, although it could be written in at least 448 ways, theoretically, in a large annotated dataset this expression is a strong feature since it is a common expression thus enables a ML classifier for instance to learn that this pattern leads to negative text.

The rest of the sentences do not contain an obvious negative sequence of words or common expressions. We know they present negative sentiment, because of our cognitive understanding of the natural language. If training a ML classifier on bigram or trigram features, two or three word sequences, from text could teach it to classify such text with no common negative words or expressions, how big the annotated Arabizi data should be to suffice such training given the high degree of lexical sparsity.

Second, as we can see from the mentioned example that negative words keep negative words company. Relatively, positive words keep positive words company. As such, if we used the lexicon-based approach with SenZi to classify a new data set that included this comment. Let's assume that SenZi classified it as negative because of one correctly classified negative word, *bankrupted* for example.

do they still dare to meet? It seems that they have nothing to do outside the government?..they are sticking on the chairs to receive imaginary incomes from a bankrupted government! … what is it they want to talk about? !! Coffee cup reading! (Fortune telling).

Then using the lexicon-based classified text to train a ML classifier might teach the classifier implicitly that *sticking on the chairs* or *dare to meet* is a negative sequence of words. Although this is approach is not recommended for classification because of overfitting the ML classifier, it might be a trick to find sentiment expressions and phrases automatically from unlabelled data.

7.5 Chapter Summary

In this chapter we introduced the lexicon-based approach for sentiment analysis. We addressed RQ2 and RQ3 by evaluating the sentiment analysis performance of the SenZi lexicon and its expansions that are developed throughout Chapters 5 and 6.

The lexicon-based approach classifies text into positive and negative sentiment classes, hence we followed two evaluation methods:

1. Randomise a sentiment class for unclassified tweets.
2. Report the classified tweets and focus the results on them.

We analysed and compared the results among the original SenZi lexicon and its expanded versions. We achieved a classification coverage of 63% of the tweets with an F1-score of 78% in one of the expanded versions of SenZi pushing the classification coverage of the original SenZi by 40%. In another expansion we achieved a classification coverage of 56% with an F1-score of 85%.

We then analysed the errors from the classified tweets. We showed the strengths and weaknesses of SenZi and the lexicon-based approach in sentiment classification of Arabizi. We traced and explained the wrongly classified and the unclassified words that lead to the mentioned results.

To the best of our knowledge, this work presents the first sentiment analysis over Arabizi text without prior transliteration attempts to Arabic.

IV. Ending

8 Conclusion

قال لها:

النساء هُنَّ الدواهي والدّواهُنَّ لا طيب للعيش بلا هُنَّ والبلا هُنَّ

فأجابت:

والرجال هم المرهمُ والمُرُّهُمُ لا طيب للعيش بلا هُمُ والبلا هُمُ

In this thesis we focused on resourcing a low-resourced heterogenic language for the task of sentiment analysis. Arabizi, a written language that came to exist out of the digital communication naturally without a standard orthographic system. It is the Latinization of the spoken dialectal Arabic, an Arabic that is derived from MSA but influenced by foreign languages, an Arabic that is esoteric to every region with different choice of words, phonemes, morphology, pronunciation, and tempo.

The main goal that drives this thesis is to reach the ability to analyse sentiment from Arabizi text directly without prior attempts of transliterating the complex script. An automatic classification of input Arabizi text into positive and negative classes. To reach this goal we proposed and tackled the following research questions:

I. *How frequent is Arabizi on social media and what makes it a challenge for sentiment analysis?*

II. *How could an Arabizi lexicon be developed and used for sentiment analysis?*

III. *Could word embeddings enhance the performance of Arabizi sentiment analysis?*

We now present a summary of our work in addressing these questions. It includes a brief summary about the challenges of the work, the methodology adopted to overcome these challenges, the resources we built, and the findings we have reached. We then present the list of contributions, our future work directions, and finally end this chapter by drawing some conclusions.

8.1 Summary

8.1.1 Foundation

One of the elements that motivated us to start this research came from the observation that the use of Arabizi has stretched out from private mediums such as mobile phone texting into public platforms like the social media. The use of Arabizi on social media lead us to ask how frequent is Arabizi used on social media and why has it been overlooked in the literature of Arabic sentiment analysis though it is popular among the Arab youth (Chapter 1).

In Chapter 2 we initiated this thesis with a pilot study about the use of Arabizi among other languages on Twitter across two Arab regions: Lebanon and Egypt.

We collected and analysed two Twitter data sets from Lebanon and Egypt in 2016. We found that the percentage of Arabic to Latin script tweets was 47% to 53% in Lebanon and 70% to 30% in Egypt.

We manually annotated two 5K Twitter datasets from the Latin script tweets, one from each of country. We found that Arabizi comprises 9.3% of Lebanon's and 19% of Egypt's Latin script tweets.

Several research in Arabic sentiment analysis (Chapter 3) reported that they have completely discarded Arabizi from their datasets prior to their sentiment analysis experiments. This motivated us to investigate and identify the linguistic issues of Arabizi that pose challenges for NLP processing and sentiment analysis (Chapter 2).

The informal texting on social media presented some linguistic deviations from the formal use of languages. Deviations that include social media abbreviations, coining new terms, typographical errors, exaggeration, in the form of repeated letters, shouting in the form of upper casing the text, expressing emotion through emojis and emoticons. Although such informal texting presents a challenge for NLP processing and analysis in any language, we found that Arabizi tops these challenges with three distinctive characteristics:

1. Richness in morphology.
2. Inconsistency in orthography.
3. Codeswitching.

Since Arabizi is a transcription of the dialect Arabic, it naturally inherits the rich inflectional and derivational morphology of Arabic. It is normal for an Arabic word to derive a hundred inflections. Inflections that consist of addition of letters, affixes, or even reduction of letters. Unlike the morphology of Turkish where the inflections are known sequences of suffixes that attach to words, Arabic inflectional derivation includes infixes where the structure of words change. Hence stemming Arabic is considered a complex task (Chapter 2).

In addition, since dialectal Arabic is a spoken, non-written, language, people Latinise Arabic in text based on some standardised letters and more on their own interpretation of spelling which has led to an inconsistent orthography.

Arabizi is more common among the bilingual youth in the Arab world. Since it is an informal texting language, it did not stop them from expressing their multilingual mixed speech while texting. It is a Latin script based language, thus there is no need to switch the script on the digital keyboards to codeswitch with English or French, the Arabs' major second languages. Hence, it became feasible for Arabs to express their dialectal and codeswitched speech in digital texting easily. The frequency of codeswitching Arabizi with English or French varies among regions and individuals.

Sentiment analysis aims at classifying text into sentiment classes automatically, positive, negative, and neutral. Sentiment analysis exist in different approaches as shown throughout the thesis. The common-ground concept among different approaches to classify text into classes is the classification of words that make up the text. Intuitively, positive words make a positive text and negative words make a negative text.

Since sentiment analysis deals with classifying words in a language, the mentioned distinctive characteristics of Arabizi present a challenge that gets in the way of sentiment analysis: lexical sparsity.

The linguistic synergy among the rich morphology, inconsistent orthography, and codeswitching in Arabizi makes it naturally high in lexical sparsity.

If a sentiment Arabizi word may be inflected in hundred forms, where each form may be transcribed in ten different spellings. This results in a thousand form for a single Arabizi word.

How can any sentiment analysis approach cope with this degree of sparsity?

8.1.2 Resources

We adopted the lexicon-based approach for the course of research presented in this thesis as a first step towards the sentiment analysis of Arabizi. This requires data resources for building the sentiment lexicon and evaluating it.

To the best of our knowledge, there are no known sentiment annotated datasets, sentiment lexicons, or published corpora for Arabizi, hence this makes Arabizi a low-resourced language. With this shortage of Arabizi data, in Chapter 4 we resourced Arabizi to meet the requirement of sentiment analysis. We chose to resource Lebanese Arabizi as the case dialect for this research.

We collected and annotated an Arabizi dataset of 1.6K tweets for the sentiment analysis evaluation. We then collected an Arabizi corpus of 1M Facebook comments to be used for building the sentiment lexicon.

We planned to build a new sentiment lexicon that deals effectively with two out of the three challenging distinctive Arabizi characteristics that resulted in high lexical sparsity, the rich morphology and the inconsistent orthography.

The lexicon creation plan consisted of two stages, lexicon generation and expansion

In the first phase, in Chapter 5, we generated Arabizi sentiment words originating from external resources that include English sentiment lexicons and a Lebanese dialect word list. First, we automatically translated English sentiment lexicons to Arabic. Then with the help of some Lebanese native students, we manually selected Lebanese sentiment words and transliterated the selected words to Arabizi. This resulted in a new Arabizi sentiment lexicon consisting of 2K sentiment words (607 positive and 1.4K negative). We named the lexicon SenZi.

In the second phase, in Chapter 6, we used a deep learning technique to extract the naturally written inflectional and orthographic forms of the words in SenZi from the collected Arabizi corpus (1M Facebook comments).

We exploited the power of word embeddings of transforming an unsupervised text into a space of word vectors. Word embeddings models align the word vectors into semantically and/or syntactically related clusters. We converted the Arabizi corpus into an embedding space of word vectors to extract the words that neighbour the SenZi words.

We proposed six different expansions of SenZi using different embedding models with different configurations, with and without word filtering. This expansion technique retrieved the inflectional and orthographic forms of the SenZi's sentiment words. The minimum expansion expanded SenZi to 9.7K words and the maximum expansion expanded it to 292.7K words.

At this point the newly generated and expanded Arabizi sentiment lexicons were ready to be evaluated using the lexicon-based sentiment classification against the prepared annotated dataset.

8.1.3 Evaluation

After handcrafting a new Arabizi sentiment lexicon and expanding it using the word embeddings, we presented a sentiment analysis evaluation. In this part we addressed RQ2 and RQ3 to understand the value of the sentiment lexicon that we created and whether expanding it with its nearest word neighbours in the embedding space improves the sentiment classification.

The methodology we followed in the course of this thesis from creating the datasets to the sentiment lexicon is designed to fit our proposed approach, lexicon-based sentiment analysis. This approach searches every word in an input text in the lexicon. It considers the text positive if it matches positive words, negative if it matches negative words. If it matched positive and negative words, then the class of the higher number of words dominates. Otherwise, if it did not match any sentiment words or an equal number of positive and negative words it leaves the text unclassified.

We presented this evaluation in Chapter 7. We followed a common binary classification method, positive and negative classes. We balanced the dataset, 800 positive and 800 negative tweets, and prepared it for the evaluation.

We present a summary of the lexicon-based approach results using SenZi and three of its best-scoring expansions in Table 8.1 below:

Lexicon	Word Size	Classified Tweets	Results Over Classified Tweets			
			R	P	F	A
SenZi Original	2K	23%	0.95	0.81	0.88	0.87
SenZi FT	35.8K	69%	0.66	0.72	0.69	0.72
SenZi Large	292.7K	63%	0.84	0.73	0.78	0.77
SenZi FT-CLSR	27.9K	56%	0.94	0.78	0.85	0.83

Table 8.1: Summary of Evaluation Results

As such we concluded the following points from the evaluation:

1. The lexicon-based approach proves to comply with Arabizi data for sentiment analysis.

2. The high degree of lexical sparsity in Arabizi gets in the way of sentiment analysis classification.
3. Using word embeddings to retrieve forms and related words of SenZi results in significant improvement in sentiment analysis.

Following the results, we presented an error analysis that reveals the limitations of the sentiment lexicons and the lexicon-based approach from the unclassified and wrongly classified tweets.

We did a word-level and sentence-level error analysis highlighting the major errors below:

Word-level:

- Unclassified Sentiment Words:
    1. Words written in different form than the ones in SenZi
    2. Codeswitched sentiment words that are written in informal English.

- Wrongly Classified Words:
    1. Retrieving an irrelevant word in the automatic expansion of SenZi.
    2. Classifying sentiment words that had different meaning in different contexts.
    3. Classifying positive words that were used in sarcasm.

Sentence-Level: Unclassified sentences contained common expressions or natural phrases that expressed sentiment without the use of sentiment words.

In Chapter 7, we discussed some of these limitations and proposed research ideas to address them. We now list and describe our contributions.

8.2 Contributions

Arabizi is a low-resourced language for NLP yet it is common among the Arab youth within some Arab regions (Chapter 1). Researchers who studied sentiment analysis for Arabic either didn't consider Arabizi for their study or attempted to transliterate it to Arabic script (Chapter

3). To the best of our knowledge, works that mentioned transliterating Arabizi to Arabic prior to sentiment analysis did not present a rigor evaluation of the transliteration (Chapter 3).

The way Arabs bridged the phonemes and syntax of Arabic with Latin script from their personal Latinisation interpretation without a consensus on an orthography introduced word ambiguity. A direct mapping of Latin script with Arabic script in an attempt to de-Latinise it (transliterate) produces incorrect Arabic words (Chapter 2).

On another end, Arabizi reflects the spoken dialectal Arabic, hence de-Latinising it, at best results in dialectal Arabic script, which is inconsistent in orthography and low-resourced as well.

We started this thesis by explaining the underlying issues of the inconsistent Latinisation of Arabic script and how they hinder the transliteration of Arabizi (Chapter 2). For that we took a different direction for analysing sentiment from Arabizi, we proposed to deal with Arabizi as a new stand-alone language independent of Arabic. Hence, our main contributions in this thesis can be summed in the following categories: Insights, Resources, Approaches, and Findings. We describe each category below. Finally, we discuss how this work contributes to the literature of Arabic NLP.

8.2.1 Insights

Considering Arabizi a new language independent but coexisting with Arabic on social media, we presented a pilot study about the percentage of Arabizi usage in comparison with Arabic and English on Twitter across Lebanon and Egypt. A fruitful insight about the demographics of Arabizi to Arabic and English usage within those Arab countries during 2016.

We then detailed some characteristics about the phonology and orthography that are unique to Arabic among English and other Latin script languages. We followed this by an investigation on how Arabizi users from different Arab regions transcribe the unique Arabic phonemes and orthography in Latin script which lead to the inconsistent orthography.

Contributing such information about the complex nature of Arabizi and the pitfalls of transliterating it should hopefully benefit any upcoming planned research about studying, transliterating, or analysing Arabizi.

8.2.2 Resources

The scarcity of NLP resources for Arabizi makes the outcome resources from this research a major contribution of this work.

1. Arabizi Datasets: We collected and preprocessed 30K Latin script tweets from Lebanon for the following annotation tasks. Three Lebanese native students annotated the dataset for: Arabizi / Not Arabizi and the Arabizi ones for sentiment: Positive, negative, and neutral.

2. Arabizi Corpus: We collected 2.2M Latin script Facebook comments from public pages from Lebanon. We then automatically identified the Arabizi comments resulting in a corpus of 1M Arabizi comments.

3. Sentiment Lexicons: We handcrafted a new Arabizi sentiment lexicon (SenZi) consisting of 2K words. We then created six expanded versions of SenZi enriching it with semantically and syntactically related words such as their orthographic and inflectional forms reaching up to 292.7K words in one of the expansions.

We made all outcome resources public, on the project's webpage[37], for the NLP and Linguistics communities for future research efforts that may include the following:

- Benchmarking efforts in Arabizi identification and Arabizi sentiment analysis.
- Creating larger Arabizi sentiment-annotated datasets quicker, since Arabizi could be identified using the publicised Arabizi identification method and dataset.

[37] https://tahatobaili.github.io/project-rbz/

- Creating parallel datasets for translation and transliteration training and evaluation among Arabizi, Arabic, and English.
- Training language models and classifiers.
- Testing different sentiment analysis approaches on the datasets.
- Using the lexicon to conduct other sentiment analysis experiments.
- Using the lexicon as a seed of words to induce more sentiment words.
- Using the lexicons as building blocks for deriving new sentiment resources for similar Levantine dialects such as Palestinian or Syrian.
- Transliterating the sentiment lexicons into Lebanese Arabic.
- Training new word embeddings from the Arabizi corpus.
- Training other Deep Learning approaches on the corpus for various downstream NLP tasks.
- Creating multi-lingual word embeddings from three corpora (Arabic, English, and Arabizi) for translation, transliteration, topic classification, word completion, text simplification, etc.
- Parsing the Arabizi corpus, or part of it, to create an Arabizi Treebank with relations and entities.
- Studies on Sociolinguistics, Dialectology, and Psycholinguistics.
- Studies on regional Bilingualism and Codeswitching.

8.2.3 Approaches

As mentioned earlier, we treated Arabizi as a new language independent of Arabic. On that basis, we developed some resources for Arabizi including a new sentiment lexicon. Our contribution on this end is the outcome resource but also a method for developing and expanding the resource.

We created the sentiment lexicon in two phases, generation and expansion, detailing every step to make it clear and easy for replicability onto other Arabic dialects or low-resourced languages especially the morphologically rich ones.

Although word embeddings was the main element in expanding SenZi, we layered the nearest word neighbours extraction with a heuristic approach to select the most syntactically relevant words. A layer that consists of normalisation and consonant letter sequence (CLS) matching. This approach proved affective for retrieving the orthographic and inflectional forms of the words in Arabizi. Then we used the same approach separately, without word embeddings, which resulted in a large expansion that increased the number of relevant forms per each SenZi word.

Researchers in Arabic NLP may take advantage of this approach in efforts on morphological analysis that could be used in lexicon generation, stemming, or text simplification.

8.2.4 Findings

Conducting sentiment analysis research on Arabizi as a low-resourced language, we have reached a classification coverage of 69% and classification results of 85% F1-score at best, using two new SenZi lexicons.

First, through this thesis we have set a new baseline of sentiment analysis for the Lebanese dialect Arabizi in the literature of Arabic NLP. A baseline that other researchers may benchmark their efforts against and build upon.

Second, we presented an empirical evidence of how enriching sentiment lexicons with word forms in a morphologically rich and orthographically inconsistent language leverages the sentiment analysis results, pushing the classification coverage by 40% for the case of Lebanese dialect Arabizi.

Third, we presented a detailed word-level error analysis of our sentiment classification results. We introduced and explained every limitation we encountered to present the major factors that get in the way of lexicon-based sentiment analysis for Arabizi. We finally proposed potential approaches to address these major limitations.

8.3 Discussion

In this research we backed up our claim that Arabizi is common in social texting by presenting an analysis of Twitter datasets extracted from the regions of Lebanon and Egypt in Chapter 2. Though Arabizi is found to be more common in mobile texting (Chapter 2) we found 6% of Twitter's data to be Arabizi across Lebanon and Egypt, that being said yet there are several works in the literature of Arabic sentiment analysis that overlooked Arabizi, some filtered it out from the Arabic datasets while others called it noise (Duwairi & Qarqaz, 2014), (Al-Kabi, et al., 2013), (Al-Kabi, et al., 2014). That being said we decided to designate a whole research on the sentiment analysis of Arabizi. A domain that has not been explored thoroughly in the literature yet it is NLP challenging for its scarcity of resources and linguistic complexities. A social language that breaks the norms of linguistic formality and structure, a multilingual language in its nature, a language unlike written languages it lacks a standard orthography. Whilst to the best of our knowledge this marks the first sentiment analysis work to address Arabizi in its natural Latin script form, we highlight how our methodology contributes to the literature.

Early in this research (Chapter 2) we scrutinised the transcription of Arabizi going through the underlying complexities in detail of how people map their dialectal phonemes of Arabic in Latin script without following a standard orthography. We discussed the linguistic challenges that this form of social Latinisation pose for text processing and sentiment classification. By that, unlike the literature (Chapter 3), we would have defined a solid nitty-gritty background about Arabizi, a rich introduction hopefully valuable for any upcoming linguistics or NLP research about Arabizi.

If Arabizi is considered low in resources, Lebanese dialect Arabizi, to the best of our knowledge, lacks NLP resources. Most NLP works on Arabizi targeted the Egyptian dialect and more recently North African with majority on Algerian dialect (Chapter 3). One of the highlights of our resource contribution is the Lebanese dialect, thus leveraging the Arabic NLP with a major Levantine dialect.

Most researchers in the literature of Arabic NLP viewed Arabizi as a transliteration challenge and worked towards that, even few works that study sentiment analysis for Arabic attempt to transliterate Arabizi automatically such as (Al-Aziz, et al., 2011), (Mataoui, et al., 2016), and (GUELLIL, et al., 2018). None of these works however presented an evaluation of the transliterated text. We on the other hand anticipated the scarcity of Lebanese Arabic resources and the difficulty to develop an Arabizi transliteration system. We considered Arabizi a new low-resourced language that happened to be rich in its linguistic complexities and worked on that basis to resource it for sentiment analysis.

The main contribution of this work lies in the development of a new Arabizi sentiment lexicon. As we've seen in Chapter 3, most efforts in developing Arabic sentiment lexicons Sifaat (Abdul-Mageed & Diab, 2012), SANA (Abdul-Mageed & Diab, 2014), ASWN (Alhazmi, et al., 2013), ArSenL, (Badaro, et al., 2014) and SLSA, (Eskander & Rambow, 2015) build upon existing sentiment labelled datasets or other sentiment lexicon such as Senti and Arabic WordNets (Esuli & Sebastiani, 2007), (Black, et al., 2006), both of which are non-existent for the Lebanese dialect Arabizi. As for Arabizi, we took a different direction in the lexicography. We approached each of the following challenges separately, first generate Arabizi sentiment words, then expand (match) them with their morphologic and orthographic variants. As such, similar to the mentioned works in the literature we also utilised translation and partial handcrafting from previous resources to generate Arabizi sentiment words, however we coupled word embeddings with a rule-based approach to expand the generated words with their variants. This simplifies the maintainability of the lexicon; it can be easily updated with new lemmas as a primary step and expanded automatically as a secondary step.

8.4 Future Work

Most research in the Arabic NLP have focused on Modern Standard Arabic; recently we have seen an increase of efforts for the dialectal Arabic. However, Arabizi is overlooked in the literature with few works that focused on transliterating it into dialectal Arabic (Chapter 3).

Since we took a different direction to deal with Arabizi and planned to resource it for sentiment analysis, one of the main challenges we faced was the high degree of lexical

sparsity which got in the way of lexicon-based sentiment analysis. For that, we proposed the third research question.

RQ3: *Could word-embeddings enhance the performance of Arabizi sentiment analysis?*

The research we presented throughout the course of this thesis showed that word embedding is a powerful approach in retrieving relevant words to address the sparsity challenge. However, we did not exploit the power of word embeddings fully yet.

If we take a step backwards and review the concept of transliterating Arabizi to Arabic. We mentioned in Chapter 3 how researchers who took this direction attempted to transliterate Arabizi by generating the Arabic equivalence of the Arabizi text computationally using rule-based approaches. We also explained how Arabizi is ambiguous by nature, thus transliterating it computationally fails in many cases (Chapter 2).

We now look into transliteration, but from a deep learning perspective. We propose the following question:

Could cross-lingual word embeddings be used to transliterate Arabizi to Arabic?

The word embeddings produce we have seen so far is a nearest word neighbour extraction from an embedding space trained on an Arabizi corpus. This embedding space is called monolingual word embedding, theoretically, because it is an embedding space of one language, Arabizi.

Cross-lingual word embeddings (CLWE) is an embedding space induced from two or more monolingual word embeddings. The goal is to align two monolingual word embeddings to induce a new CLWE where the word vectors of similar meanings align next to each other within both languages. Figure 8.1 shows two monolingual word embeddings being aligned, English and Italian.

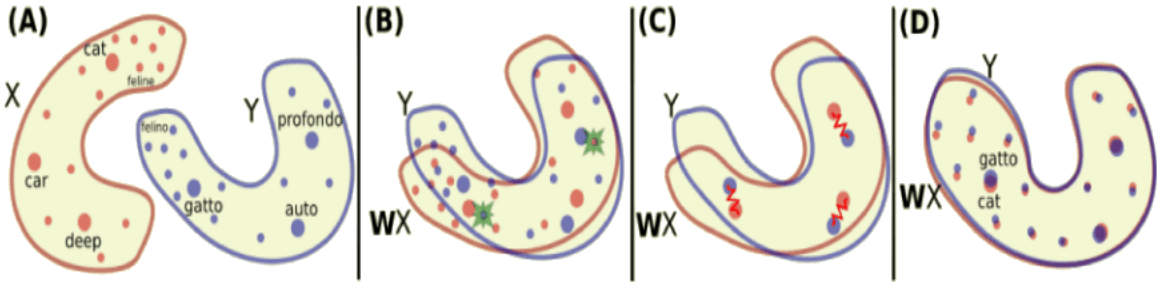


Figure 8.1[38]: Multi-Lingual Word Embeddings

One way of aligning the embedding spaces is to create a translation matrix of the two languages and keep tuning its parameters so that multiplying a word vector from one language, *cat* for example, by the matrix parameters give a word vector closest in meaning to that word (*cat*) in the target language, *gatto* in this case.

The newly induced aligned CLWE would have the vectors of *cat* and *gatto* close in distance in that space. As such aligning an Arabizi embedding space with an Arabic embedding space opens up a new perspective for transliteration. A transition from generating a list of possible transliterations for every Arabizi word computationally into finding a naturally written Arabic word that is closest to the Arabizi word in the embedded vector space.

Throughout the research conducted in this thesis, we have demonstrated an impactful use case of word embeddings to resource Arabizi and proposed to use it for transliteration to Arabic as well. Latinisation however is not exclusive for Arabic. As such, our second future research question would be:

Could we use word embeddings to resource or transliterate other Latinised languages?

We mentioned in the pilot study, on the usage of Arabizi percentage in social media (Chapter 2), that among the collected Latin script comments from Lebanon and Egypt, we identified

[38] Image: https://www.techleer.com/articles/451-muse-multilingual-unsupervised-and-supervised-embeddings/

Latinised Hindi and Filipino. This phenomenon is also common for Greek[39], Farsi, Urdu, Hindi, Bengali, Telugu, and other non-Latin script languages[40]. The amount of Arabizi data generated in digital texting and social media might be very small in comparison to the widely spoken Far Eastern languages. Hence, the contributions presented in this thesis from methods and findings on resourcing Arabizi to the use of word embeddings in retrieving naturally written related word forms of Arabizi motivates us to explore similar directions in other Latinised languages. If researching CLWE succeeds in bridging Arabizi with Arabic for automatic transliteration, then this could potentially open several directions to bridge other Latinsied languages with their original script as well. However, every language has its semantic, syntactic, orthographic, morphologic, phonologic, and morphononemic structure as such transferring approaches into new language domains should be preceded by a linguistic investigation of the target language. Word embeddings for instance might need tuning in the parameters to capture semantic similarities in Telugu, according to the structure of the language. Parameters might include sentence, word, or character level information such as sequence of letters, morphemes, stems, etc.

8.5 Conclusions

As the thesis comes to an end, we reflect upon some observations and draw some conclusions.

First, our initial plan to address sentiment analysis for Arabizi was to use the recent approaches that achieved state of the art results in sentiment analysis for the English language (Chapter 3). The superlative data-driven approaches from ML classification to neural network transformers have leveraged the science and applications of NLP significantly over the last few years. However, with the lack of resources for Arabizi, we were regressed behind the sentiment analysis state of the art.

[39] https://en.wikipedia.org/wiki/Greeklish
[40] https://www.open.edu/openlearn/education/educational-technology-and-practice/educational-practice/hinglish-pinglish-binglish-minglish

Throughout our study of Arabizi we identified the underlying linguistic complexities of natural Latinisation of dialectal Arabic to discover that the approaches that leveraged NLP for English will most likely be obstructed for Arabizi.

I. What is considered a state of the art approach in NLP is not necessary the case for languages that are phonologically, morphologically, and orthographically different than English.

Second, from our study of Arabizi we knew that the inconsistent orthography and inherited rich morphology of Arabizi makes this language highly sparse. However, after mining the Facebook corpus for matching SenZi with relevant words, the number of forms many words retrieved was beyond our expectation. Some words reached 2.7K forms. Thus, we learned that before conducting sentiment classification or word classification related tasks, we had a major challenge to address first.

II. If a low-resourced language has one of these characteristics *richness in morphology* or *inconsistency in orthography*, it presents a high degree of lexical sparsity. However, undertaking NLP research in a language that has both of these characteristics, one has to be aware of the multiplied effect they have on the lexical sparsity.

Finally, we have learned that sentiment analysis is not limited to the sense of words. We found that even strong sentiment words might have different meaning in different contexts. Positive and negative sentiment, love or hate, is often expressed in phrases that lack sentiment words as well.

III. With all the value a sentiment lexicon brings to AI, our human cognitive ability to interpret sentiment from languages surrounding our sphere cannot be coped in sentiment lexicons.

Bibliography

Abbasi, A., Chen, H. & Salem, A., 2008. Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. *ACM Transactions on Information Systems (TOIS),* Volume 26, p. 12.

Abdulla, N. A., Ahmed, N. A., Shehab, M. A. & Al-Ayyoub, M., 2013. *Arabic sentiment analysis: Lexicon-based and corpus-based.* s.l., s.n., pp. 1-6.

Abdul-Mageed, M. & Diab, M., 2012. *Toward building a large-scale Arabic sentiment lexicon.* s.l., s.n., pp. 18-22.

Abdul-Mageed, M. & Diab, M. T., 2011. *Subjectivity and sentiment annotation of modern standard arabic newswire.* s.l., s.n., pp. 110-118.

Abdul-Mageed, M. & Diab, M. T., 2014. *Sana: A large scale multi-genre, multi-dialect lexicon for arabic subjectivity and sentiment analysis..* s.l., s.n., pp. 1162-1169.

Aboelezz, M., 2009. *Latinised Arabic and connections to bilingual ability.* s.l., s.n.

Ahmed, S., Pasquier, M. & Qadah, G., 2013. *Key issues in conducting sentiment analysis on Arabic social media text.* s.l., s.n., pp. 72-77.

Al Sallab, A. et al., 2015. *Deep learning models for sentiment analysis in Arabic.* s.l., s.n., pp. 9-17.

Alabdulqader, E. et al., 2014. Computer Mediated Communication: Patterns & Language Transformations of Youth in Arabic-speaking Populations. *Information Technology & Computer Science (IJITCS),* Volume 17, p. 85.

Al-Ayyoub, M., Khamaiseh, A. A., Jararweh, Y. & Al-Kabi, M. N., 2019. A comprehensive survey of arabic sentiment analysis. *Information Processing & Management,* Volume 56, pp. 320-342.

Al-Ayyoub, M., Nuseir, A., Kanaan, G. & Al-Shalabi, R., 2016. Hierarchical Classifiers for Multi-Way Sentiment Analysis of Arabic Reviews. *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS,* Volume 7, pp. 531-539.

Al-Azani, S. & El-Alfy, E.-S. M., 2017. Using word embedding and ensemble learning for highly imbalanced data sentiment analysis in short arabic text. *Procedia Computer Science,* Volume 109, pp. 359-366.

Al-Aziz, A. M. A., Gheith, M. & Ahmed, A. S. E., 2011. Toward Building Arabizi Sentiment Lexicon based on Orthographic Variants Identification.

Al-Badrashiny, M., Eskander, R., Habash, N. & Rambow, O., 2014. *Automatic transliteration of romanized dialectal Arabic.* s.l., s.n., pp. 30-38.

Al-Fedaghi, S. S. & Al-Sadoun, H. B., 1990. Morphological compression of Arabic text. *Information processing & management,* Volume 26, pp. 303-316.

Alhazmi, S., Black, W. & McNaught, J., 2013. Arabic SentiWordNet in relation to SentiWordNet 3.0. *2180,* Volume 1266, p. 1.

Al-Kabi, M. et al., 2013. *An opinion analysis tool for colloquial and standard Arabic.* s.l., s.n., pp. 23-25.

Al-Kabi, M. N. et al., 2014. Opinion mining and analysis for arabic language. *International Journal of Advanced Computer Science and Applications (IJACSA), SAI Publisher,* Volume 5.

Al-Khatib, M. & Sabbah, E. H., 2008. Language choice in mobile text messages among Jordanian university students. *SKY Journal of Linguistics,* Volume 21, pp. 37-65.

Allehaiby, W. H., 2013. Arabizi: An Analysis of the Romanization of the Arabic Script from a Sociolinguistic Perspective. *Arab World English Journal,* Volume 4, pp. 52-62.

Al-Radaideh, Q. A. & Al-Qudah, G. Y., 2017. Application of rough set-based feature selection for Arabic sentiment analysis. *Cognitive Computation,* Volume 9, pp. 436-445.

Al-Rowaily, K., Abulaish, M., Haldar, N. A.-H. & Al-Rubaian, M., 2015. BiSAL--A bilingual sentiment analysis lexicon to analyze Dark Web forums for cyber security. *Digital Investigation,* Volume 14, pp. 53-62.

Al-Sallab, A. et al., 2017. Aroma: A recursive deep learning model for opinion mining in arabic as a low resource language. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP),* Volume 16, p. 25.

Altowayan, A. A. & Tao, L., 2016. *Word embeddings for Arabic sentiment analysis.* s.l., s.n., pp. 3820-3825.

Al-Twairesh, N., Al-Khalifa, H. & AlSalman, A., 2016. *Arasenti: Large-scale twitter-specific arabic sentiment lexicons.* s.l., s.n., pp. 697-705.

Aly, M. & Atiya, A., 2013. *Labr: A large scale arabic book reviews dataset.* s.l., s.n., pp. 494-498.

Assiri, A., Emam, A. & Aldossari, H., 2015. Arabic Sentiment Analysis: A Survey. *International Journal of Advanced Computer Science & Applications,* Volume 1, pp. 75-85.

Baccianella, S., Esuli, A. & Sebastiani, F., 2010. *Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining.*. s.l., s.n., pp. 2200-2204.

Badaro, G. et al., 2014. *A large scale Arabic sentiment lexicon for Arabic opinion mining.* s.l., s.n., pp. 165-173.

Baly, R. et al., 2017. Comparative evaluation of sentiment analysis methods across Arabic dialects. *Procedia Computer Science,* Volume 117, pp. 266-273.

Baly, R. et al., 2017. A sentiment treebank and morphologically enriched recursive deep models for effective sentiment analysis in arabic. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP),* Volume 16, p. 23.

Baly, R. et al., 2019. ArSentD-LEV: A multi-topic corpus for target-based sentiment analysis in Arabic levantine tweets. *arXiv preprint arXiv:1906.01830.*

Barbosa, L. & Feng, J., 2010. *Robust sentiment detection on twitter from biased and noisy data.* s.l., s.n., pp. 36-44.

Basis-Technology, 2012. The Burgeoning Challenge of Deciphering Arabic Chat.

Bhandari, A., 2018. *Arabizi: A Language Shaping the Youth Mindset,* s.l.: s.n.

Bhuta, S. & Doshi, U., 2014. *A review of techniques for sentiment analysis Of Twitter data.* s.l., s.n., pp. 583-591.

BIANCHI, R. M., 2012. 3arabizi-When local Arabic meets global English. *Acta Linguistica Asiatica,* Volume 2, pp. 89-100.

Bies, A. et al., 2014. *Transliteration of arabizi into arabic orthography: Developing a parallel annotated arabizi-arabic script sms/chat corpus.* s.l., s.n., pp. 93-103.

Black, W. et al., 2006. *Introducing the Arabic wordnet project.* s.l., s.n., pp. 295-300.

Bojanowski, P., Grave, E., Joulin, A. & Mikolov, T., 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics,* Volume 5, pp. 135-146.

Bollen, J., Mao, H. & Zeng, X., 2011. Twitter mood predicts the stock market. *Journal of computational science,* Volume 2, pp. 1-8.

Callison-Burch, C., Koehn, P., Monz, C. & Zaidan, O. F., 2011. *Findings of the 2011 workshop on statistical machine translation.* s.l., s.n., pp. 22-64.

Cambria, E., Poria, S., Gelbukh, A. & Thelwall, M., 2017. Sentiment analysis is a big suitcase. *IEEE Intelligent Systems,* Volume 32, pp. 74-80.

Chalabi, A. & Gerges, H., 2012. *Romanized Arabic Transliteration.* s.l., s.n., p. 89.

Church, K. W. & Hanks, P., 1990. Word association norms, mutual information, and lexicography. *Computational linguistics,* Volume 16, pp. 22-29.

Dahou, A. et al., 2016. *Word embeddings and convolutional neural network for arabic sentiment classification.* s.l., s.n., pp. 2418-2427.

Darwish, K., 2014. Arabizi Detection and Conversion to Arabic. *ANLP 2014*, p. 217.
De Roeck, A. N. & Al-Fares, W., 2000. *A morphologically sensitive clustering algorithm for identifying Arabic roots.* s.l., s.n., pp. 199-206.
Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
Diab, M. T. et al., 2014. *Tharwa: A Large Scale Dialectal Arabic-Standard Arabic-English Lexicon.*. s.l., s.n., pp. 3782-3789.
Duwairi, R. M., 2015. *Sentiment analysis for dialectical Arabic.* s.l., s.n., pp. 166-170.
Duwairi, R. M., Ahmed, N. A. & Al-Rifai, S. Y., 2015. Detecting sentiment embedded in Arabic social media--A lexicon-based approach. *Journal of Intelligent & Fuzzy Systems*, Volume 29, pp. 107-117.
Duwairi, R. M., Alfaqeh, M., Wardat, M. & Alrabadi, A., 2016. *Sentiment analysis for Arabizi text.* s.l., s.n., pp. 127-132.
Duwairi, R. M. & Qarqaz, I., 2014. *Arabic sentiment analysis using supervised classification.* s.l., s.n., pp. 579-583.
Elhawary, M. & Elfeky, M., 2010. *Mining Arabic business reviews.* s.l., s.n., pp. 1108-1113.
El-Makky, N. et al., 2014. *Sentiment analysis of colloquial Arabic tweets.* s.l., s.n., pp. 1-9.
ElSahar, H. & El-Beltagy, S. R., 2015. *Building large arabic multi-domain resources for sentiment analysis.* s.l., s.n., pp. 23-34.
Eskander, R., Al-Badrashiny, M., Habash, N. & Rambow, O., 2014. *Foreign words and the automatic processing of Arabic social media text written in Roman script.* s.l., s.n., pp. 1-12.
Eskander, R. & Rambow, O., 2015. *SLSA: A sentiment lexicon for Standard Arabic.* s.l., s.n., pp. 2545-2550.
Esuli, A. & Sebastiani, F., 2007. SENTIWORDNET: A high-coverage lexical resource for opinion mining. *Evaluation*, pp. 1-26.
Farha, I. A. & Magdy, W., 2019. *Mazajak: An Online Arabic Sentiment Analyser.* s.l., s.n., pp. 192-198.
Farra, N., Challita, E., Assi, R. A. & Hajj, H., 2010. *Sentence-level and document-level sentiment mining for arabic texts.* s.l., s.n., pp. 1114-1119.
Fernández-Gavilanes, M. et al., 2016. Unsupervised method for sentiment analysis in online texts. *Expert Systems with Applications*, Volume 58, pp. 57-75.
GIBSON, M., 2015. A Framework for Measuring the Presence of Minority Languages in Cyberspace. *Linguistic and Cultural Diversity in Cyberspace*, p. 61.

Glavas, G., Litschko, R., Ruder, S. & Vulic, I., 2019. How to (Properly) Evaluate Cross-Lingual Word Embeddings: On Strong Baselines, Comparative Analyses, and Some Misconceptions. *arXiv preprint arXiv:1902.00508.*

GUELLIL, I. et al., 2018. *Arabizi sentiment analysis based on transliteration and automatic corpus annotation.* s.l., s.n., pp. 335-341.

Guellil, I., Azouaou, F., Abbas, M. & Fatiha, S., 2017. *Arabizi transliteration of algerian Arabic dialect into modern standard Arabic.* s.l., s.n.

Habash, N., Eskander, R. & Hawwari, A., 2012. *A morphological analyzer for Egyptian Arabic.* s.l., s.n., pp. 1-9.

Howard, J. & Ruder, S., 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146.*

Hu, M. & Liu, B., 2004. *Mining and summarizing customer reviews.* s.l., s.n., pp. 168-177.

Hu, X., Tang, J., Gao, H. & Liu, H., 2013. *Unsupervised sentiment analysis with emotional signals.* s.l., s.n., pp. 607-618.

Itani, M. M., Zantout, R. N., Hamandi, L. & Elkabani, I., 2012. *Classifying sentiment in arabic social networks: Naive search versus naive bayes.* s.l., s.n., pp. 192-197.

Jaran, S. A. & Al-Haq, F. A.-A., 2015. The Use of Hybrid Terms and Expressions in Colloquial Arabic among Jordanian College Students: A Sociolinguistic Study.. *English Language Teaching,* Volume 8, pp. 86-97.

Keong, Y. C., Hameed, O. R. & Abdulbaqi, I. A., 2015. The use of Arabizi in English texting by Arab postgraduate students at UKM. *The English Literature Journal,* Volume 2, pp. 281-288.

Kiritchenko, S., Zhu, X. & Mohammad, S. M., 2014. Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research,* Volume 50, pp. 723-762.

Kouloumpis, E., Wilson, T. & Moore, J., 2011. *Twitter sentiment analysis: The good the bad and the omg!.* s.l., s.n.

Liu, B., 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies,* Volume 5, pp. 1-167.

Liu, B., 2015. *Sentiment analysis: Mining opinions, sentiments, and emotions.* s.l.:Cambridge University Press.

Liu, B. & Zhang, L., 2012. A survey of opinion mining and sentiment analysis. In: *Mining text data.* s.l.:Springer, pp. 415-463.

Maamouri, M., Bies, A., Buckwalter, T. & Mekki, W., 2004. *The penn arabic treebank: Building a large-scale annotated arabic corpus.* s.l., s.n., pp. 466-467.

Masmoudi, A. et al., 2015. *Arabic transliteration of romanized tunisian dialect text: A preliminary investigation.* s.l., s.n., pp. 608-619.

Mataoui, M., Zelmati, O. & Boumechache, M., 2016. A proposed lexicon-based sentiment analysis approach for the vernacular Algerian Arabic. *Res. Comput. Sci,* Volume 110, pp. 55-70.

May, J., Benjira, Y. & Echihabi, A., 2014. An arabizi-english social media statistical machine translation system.

Mikolov, T. et al., 2013. *Distributed representations of words and phrases and their compositionality.* s.l., s.n., pp. 3111-3119.

Mohit, B. et al., 2014. *The first QALB shared task on automatic text correction for Arabic.* s.l., s.n., pp. 39-47.

Mourad, A. & Darwish, K., 2013. *Subjectivity and sentiment analysis of modern standard Arabic and Arabic microblogs.* s.l., s.n., pp. 55-64.

Muhammed, R., Farrag, M., Elshamly, N. & Abdel-Ghaffar, N., 2011. *Summary of Arabizi or Romanization: The dilemma of writing Arabic texts.* s.l., s.n., pp. 18-19.

Nabil, M., Aly, M. & Atiya, A., 2015. *Astd: Arabic sentiment tweets dataset.* s.l., s.n., pp. 2515-2519.

Naji, N. & Allan, J., 2016. On Cross-Script Information Retrieval. In: *European Conference on Information Retrieval.* s.l.:s.n., pp. 796-802.

Nakov, P. et al., 2016. *SemEval-2016 task 4: Sentiment analysis in Twitter.* s.l., s.n., pp. 1-18.

Nanli, Z., Ping, Z., Weiguo, L. & Meng, C., 2012. *Sentiment analysis: A literature review.* s.l., s.n., pp. 572-576.

Neubig, G., 2016. Lexicons and minimum risk training for neural machine translation: NAIST-CMU at WAT2016. *arXiv preprint arXiv:1610.06542.*

Neubig, G. & Watanabe, T., 2016. Optimization for statistical machine translation: A survey. *Computational Linguistics,* Volume 42, pp. 1-54.

O'Connor, B., Balasubramanyan, R., Routledge, B. R. & Smith, N. A., 2010. *From tweets to polls: Linking text sentiment to public opinion time series.* s.l., s.n.

Pak, A. & Paroubek, P., 2010. *Twitter as a corpus for sentiment analysis and opinion mining..* s.l., s.n., pp. 1320-1326.

Pang, B., Lee, L. & others, 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval,* Volume 2, pp. 1-135.

Papineni, K., Roukos, S., Ward, T. & Zhu, W.-J., 2002. *BLEU: a method for automatic evaluation of machine translation.* s.l., s.n., pp. 311-318.

Pasha, A. et al., 2014. *Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic.*. s.l., s.n., pp. 1094-1101.

Peters, M. E. et al., 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365.*

Radcliffe, D. & Bruni, P., 2019. State of Social Media, Middle East: 2018. 1.

Refaee, E. & Rieser, V., 2014. *An arabic twitter corpus for subjectivity and sentiment analysis.*. s.l., s.n., pp. 2268-2273.

Riloff, E., Wiebe, J. & Wilson, T., 2003. *Learning subjective nouns using extraction pattern bootstrapping.* s.l., s.n., pp. 25-32.

Rosca, M. & Breuel, T., 2016. Sequence-to-sequence neural network models for transliteration. *arXiv preprint arXiv:1610.09565.*

Ruder, S., Vulić, I. & Søgaard, A., 2017. A survey of cross-lingual word embedding models. *arXiv preprint arXiv:1706.04902.*

Saif, H., Fernández, M., He, Y. & Alani, H., 2014. On stopwords, filtering and data sparsity for sentiment analysis of twitter.

Saif, H., Fernandez, M., He, Y. & Alani, H., 2014. *Senticircles for contextual and conceptual semantic sentiment analysis of twitter.* s.l., s.n., pp. 83-98.

Salamah, J. B. & Elkhlifi, A., 2014. *Microblogging opinion mining approach for Kuwaiti dialect.* s.l., s.n., p. 388.

Saleh, M. R., Martín-Valdivia, M. T., Montejo-Ráez, A. & Ureña-López, L. A., 2011. Experiments with SVM to classify opinions in different domains. *Expert Systems with Applications*, Volume 38, pp. 14799-14804.

Santos, C. N. d., Xiang, B. & Zhou, B., 2015. Classifying relations by ranking with convolutional neural networks. *arXiv preprint arXiv:1504.06580.*

Sembok, T. M. T., Ata, B. M. A. & Bakar, Z. A., 2011. *A rule-based Arabic stemming algorithm.* s.l., s.n., pp. 392-397.

Shaaban, K. A., 1997. Bilingual education in Lebanon. In: *Bilingual Education.* s.l.:Springer, pp. 251-259.

Shoukry, A. M., 2013. Arabic sentence-level sentiment analysis.

Shoukry, A. & Rafea, A., 2012. *Sentence-level Arabic sentiment analysis.* s.l., s.n., pp. 546-550.

Smrž, O., 2007. *Elixirfm: implementation of functional arabic morphology.* s.l., s.n., pp. 1-8.

Socher, R., Huval, B., Manning, C. D. & Ng, A. Y., 2012. *Semantic compositionality through recursive matrix-vector spaces.* s.l., s.n., pp. 1201-1211.

Socher, R. et al., 2011. *Semi-supervised recursive autoencoders for predicting sentiment distributions.* s.l., s.n., pp. 151-161.

Socher, R. et al., 2013. *Recursive deep models for semantic compositionality over a sentiment treebank.* s.l., s.n., pp. 1631-1642.

Sullivan, N., 2017. *Writing Arabizi: Orthographic Variation in Romanized Lebanese Arabic on Twitter,* s.l.: s.n.

Taboada, M. et al., 2011. Lexicon-based methods for sentiment analysis. *Computational linguistics,* Volume 37, pp. 267-307.

Taghva, K., Elkhoury, R. & Coombs, J., 2005. *Arabic stemming without a root dictionary.* s.l., s.n., pp. 152-157.

Taji, D. et al., 2018. *An Arabic morphological analyzer and generator with copious features.* s.l., s.n., pp. 140-150.

Tang, D., Qin, B. & Liu, T., 2016. Aspect level sentiment classification with deep memory network. *arXiv preprint arXiv:1605.08900.*

Tellez, E. S. et al., 2017. A simple approach to multilingual polarity classification in Twitter. *Pattern Recognition Letters,* Volume 94, pp. 68-74.

Thelwall, M., Buckley, K. & Paltoglou, G., 2012. Sentiment strength detection for the social web. *Journal of the American Society for Information Science and Technology,* Volume 63, pp. 163-173.

Thelwall, M. et al., 2010. Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology,* Volume 61, pp. 2544-2558.

Tobaili, T., 2016. Arabizi Identification in Twitter Data. *ACL 2016,* p. 51.

Turney, P. D., 2002. *Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews.* s.l., s.n., pp. 417-424.

Vo, D.-T. & Zhang, Y., 2015. *Target-dependent twitter sentiment classification with rich automatic features.* s.l., s.n.

Vulić, I. & Moens, M.-F., 2015. *Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings.* s.l., s.n., pp. 363-372.

Wang, L. & Xia, R., 2017. *Sentiment lexicon construction with representation learning based on hierarchical sentiment supervision.* s.l., s.n., pp. 502-510.

Wilson, T., Wiebe, J. & Hoffmann, P., 2005. *Recognizing contextual polarity in phrase-level sentiment analysis.* s.l., s.n.

Yaghan, M. A., 2008. "Arabizi": A Contemporary Style of Arabic Slang. *Design Issues,* Volume 24, pp. 39-52.

Yue, L. et al., 2018. A survey of sentiment analysis in social media. *Knowledge and Information Systems*, pp. 1-47.

Zhang, L. et al., 2011. Combining lexicon-based and learning-based methods for Twitter sentiment analysis. *HP Laboratories, Technical Report HPL-2011,* Volume 89.

Zhang, L., Wang, S. & Liu, B., 2018. Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Volume 8, p. e1253.

Zhou, H., Chen, L., Shi, F. & Huang, D., 2015. *Learning bilingual sentiment word embeddings for cross-language sentiment classification.* s.l., s.n., pp. 430-440.